Yandex

Yandex

WSDM 2020 hands-on tutorial

Practice of Efficient Data Collection via Crowdsourcing: Aggregation, Incremental Relabelling, and Pricing

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

Part VI

Theory on aggregation

Valentina Fedorova,
Research analyst

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

Tutorial outline

Introduction: 20 min

Part I: 40 min
Main Components

Coffee break:
30 min

Part II: 25 min
Brainstorming pipeline

Part III: 10 min
Introduction to Crowd Platform

Part IV: 85 min
Set & Run Projects

Lunch break:
90 min

Part V: 35 min
Interface & Quality control

Part VI: 25 min
Theory on Aggregation

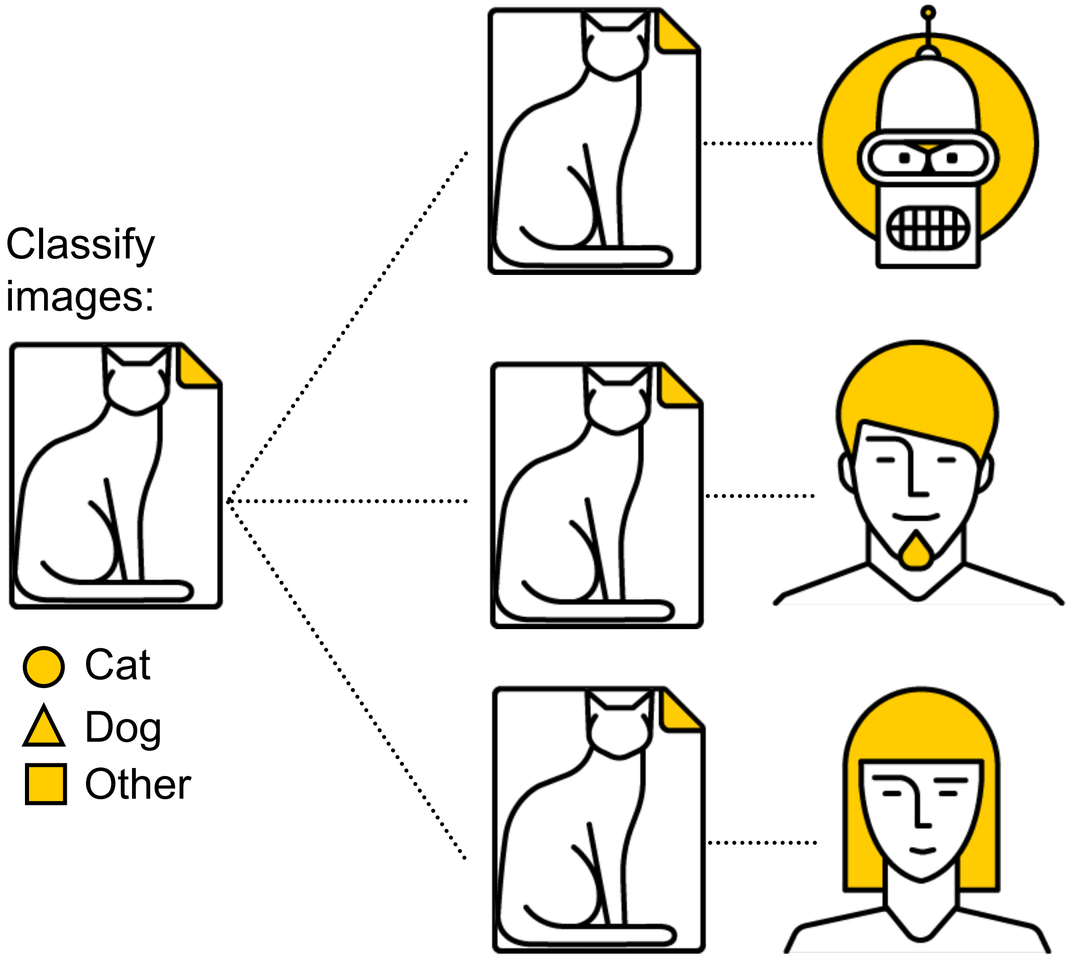
Coffee break:
30 min

Part VI: 60 min
Set & Run Projects cont.

Part VII: 20 min
Incremental relabeling and pricing

Part VIII: 10 min
Results & Conclusions

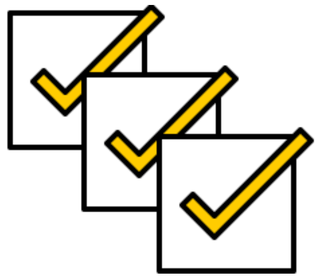
Labelling data with crowdsourcing



- How to choose a reliable label?
- How many workers per object?
- How much to pay to workers?
- …

Evaluation of labelling approaches

Accuracy



vs

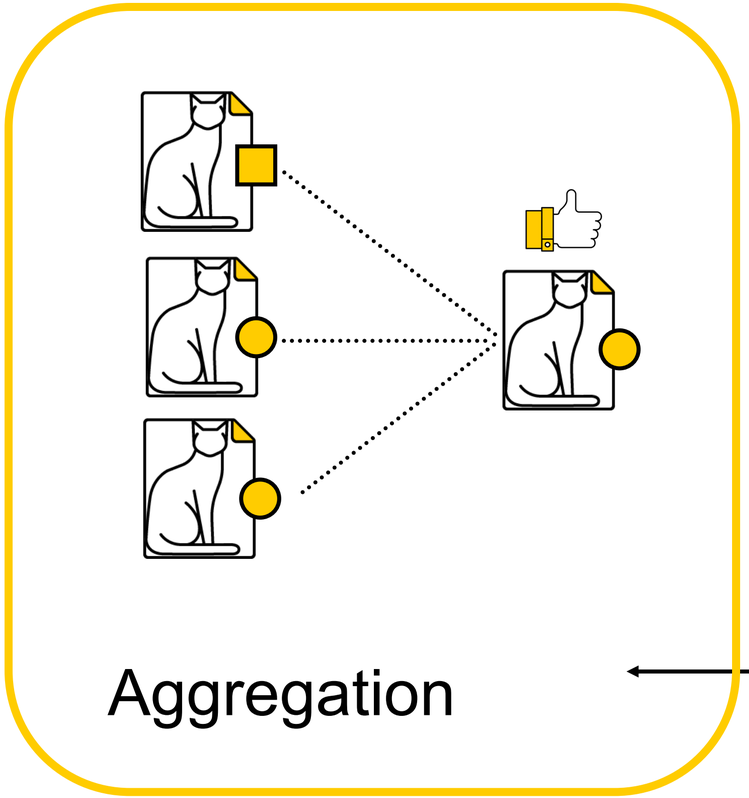
Cost

- Labels with a maximal level of accuracy for a given budget

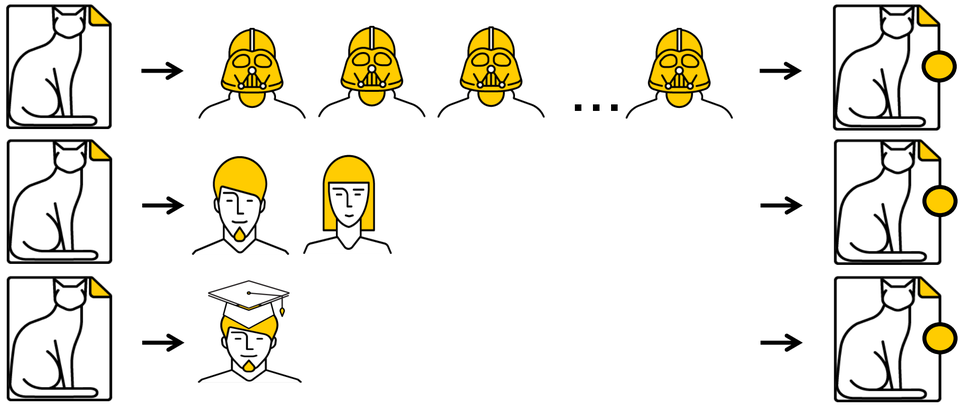
or

- Labels of a chosen accuracy level for a minimal budget

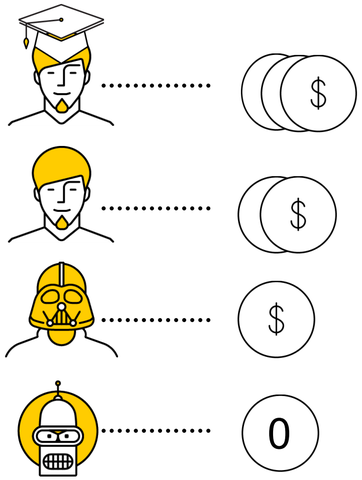
Key components of labelling with crowds



Aggregation



Incremental relabelling

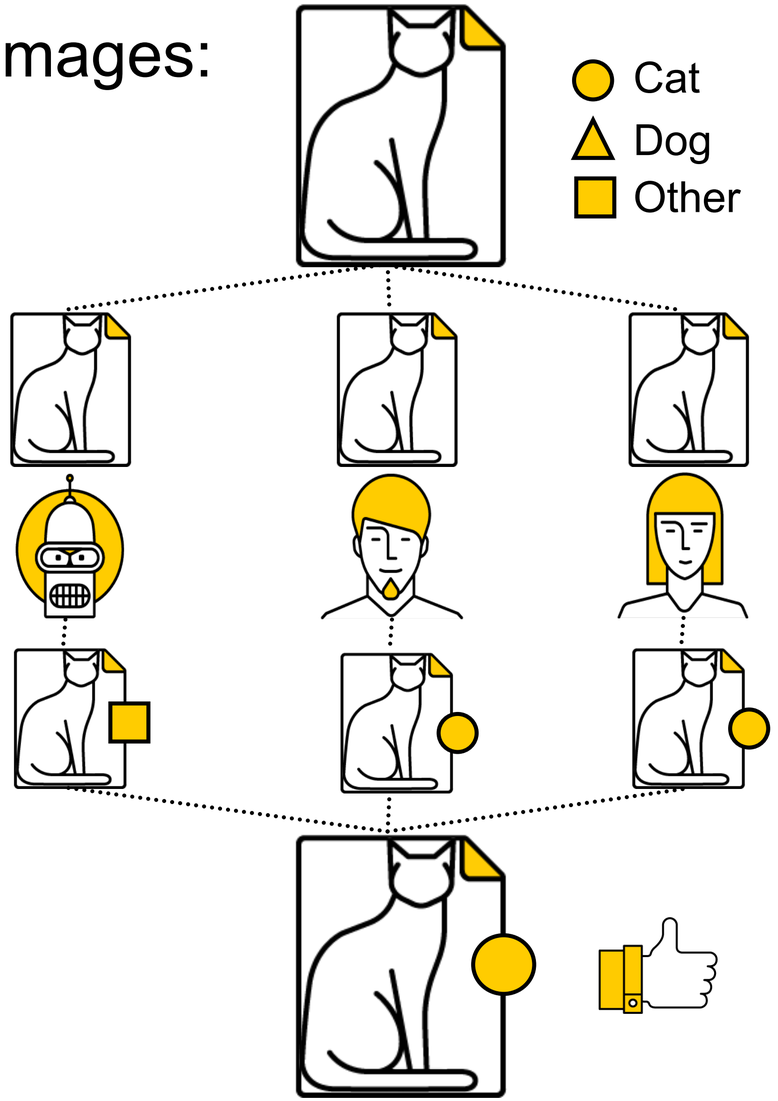


Performance-based pricing

Aggregation

Labelling data with crowds

Classify images:

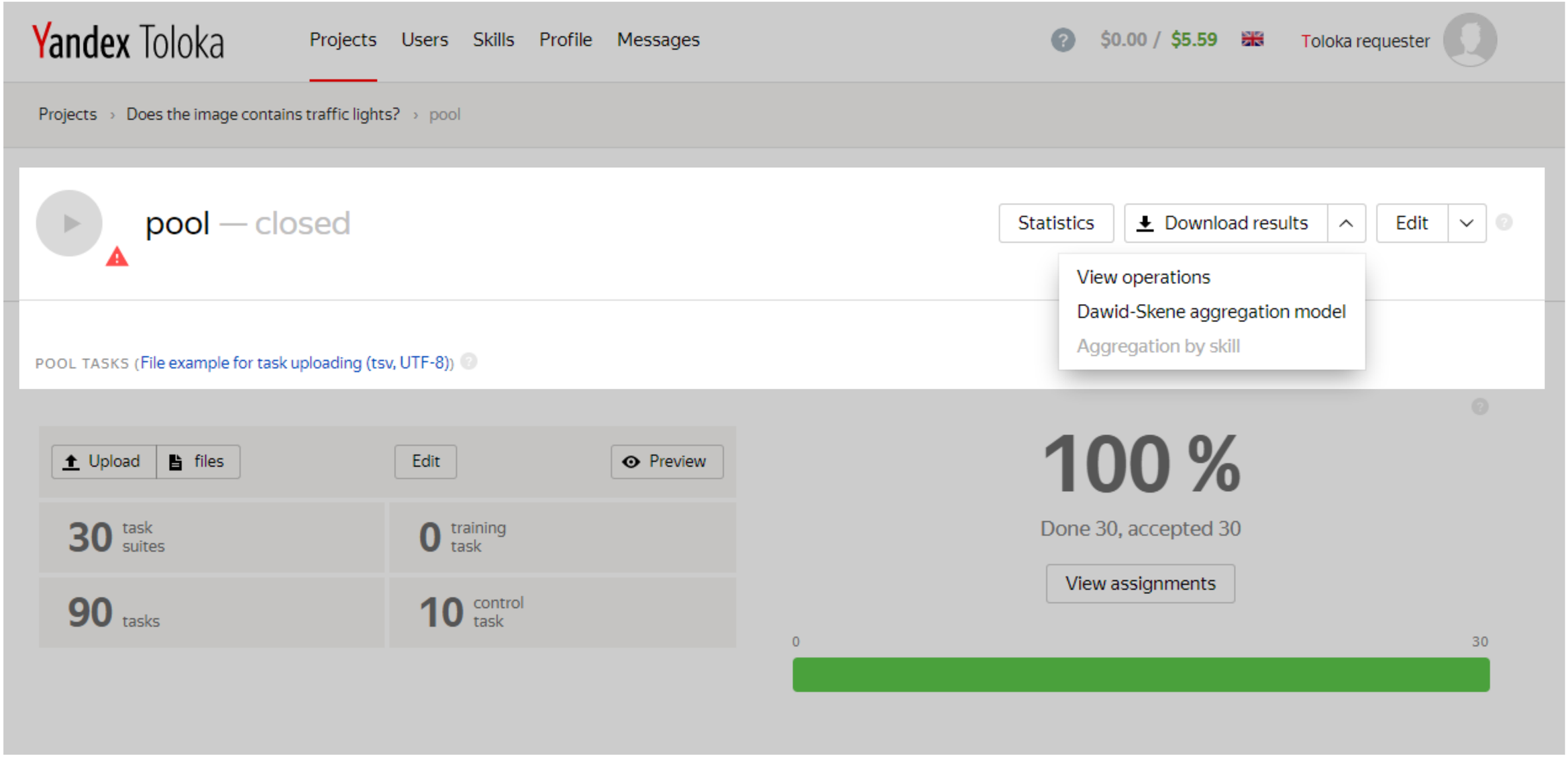


Upload multiple copies of each object to label

Workers assign noisy labels to objects

Aggregate multiple labels for each object into a more reliable one

Process results



Multiclass labels

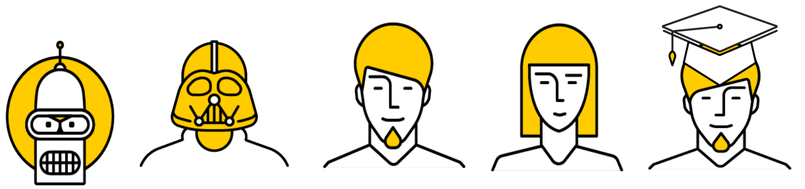
Project 1: Filter images

Are there shoes in the picture?

Yes

No

Notation

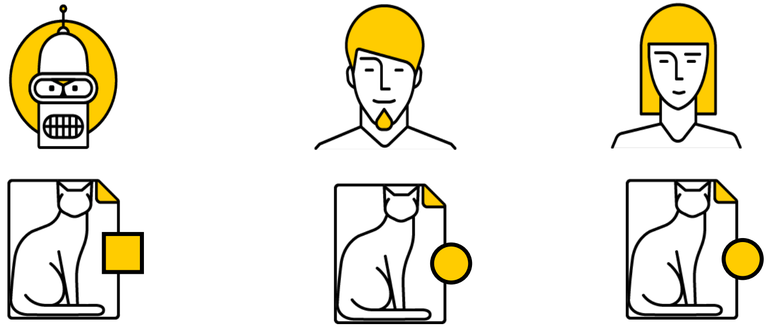
- Categories $k \in \{1, \ldots, K\}$. E.g.: ● Cat ▲ Dog ■ Other
- Objects $j \in \{1, \ldots, J\}$. E.g.: 
- Workers: $w \in \{1, \ldots, W\}$. E.g.: 
  - $W_j \subseteq \{1, \ldots, W\}$ - workers labelled object j

The simplest aggregation: Majority Vote (MV)

The problem of aggregation:

- Observe noisy labels

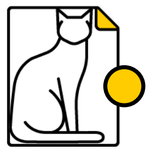
$$\mathbf{y} = \{y_j^w \mid j = 1, \ldots, J \text{ and } w = 1, \ldots, W\}$$



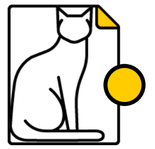
- Recover true labels $\mathbf{z} = \{z_j \mid j = 1, \ldots, J\}$

A straightforward solution:


: 1 vote

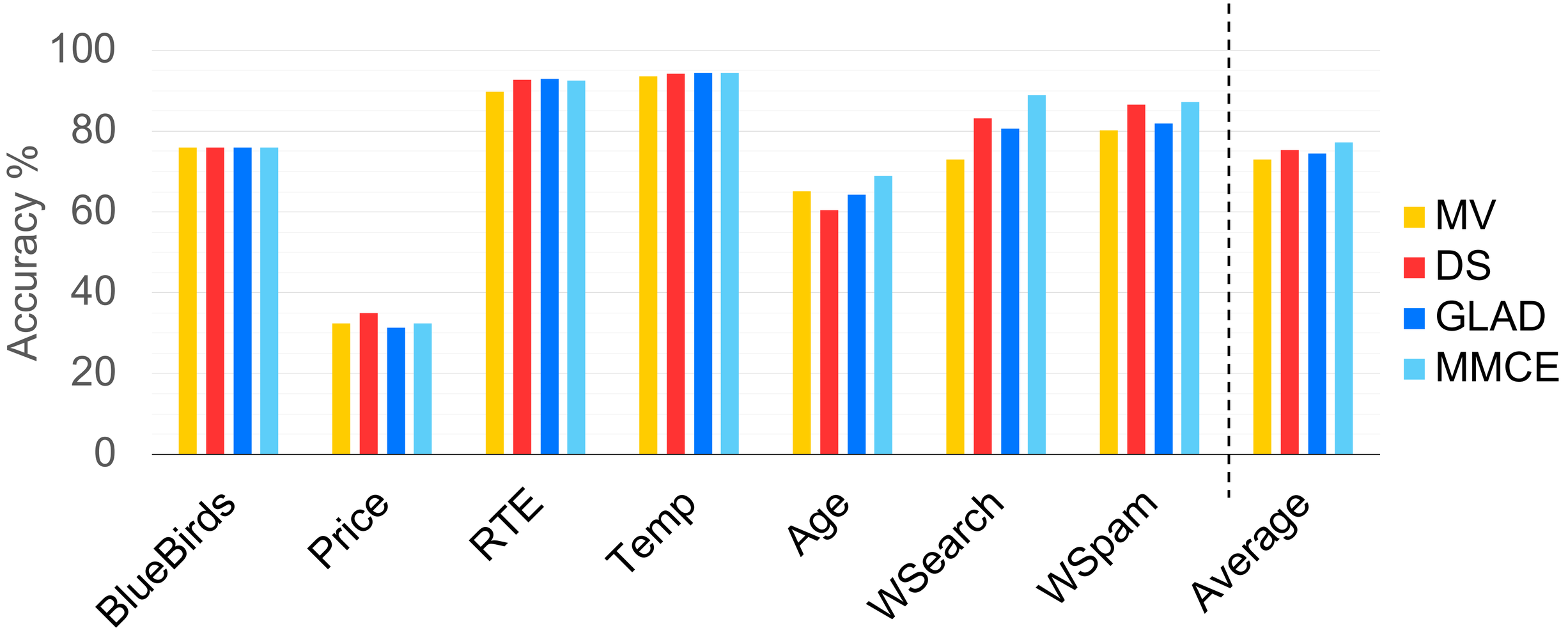

: 2 votes

⇒ MV:



$\hat{z}_j^{MV} = \arg\max_{y=1,\ldots,K} \sum_{w \in W_j} \delta(y = y_j^w)$, where $\delta(A) = 1$ if A is true and 0 otherwise

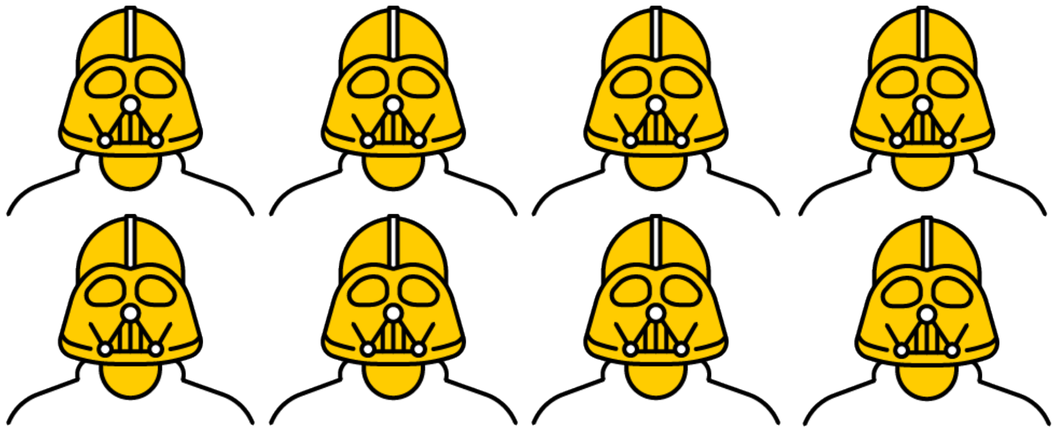
Performance of MV vs other methods



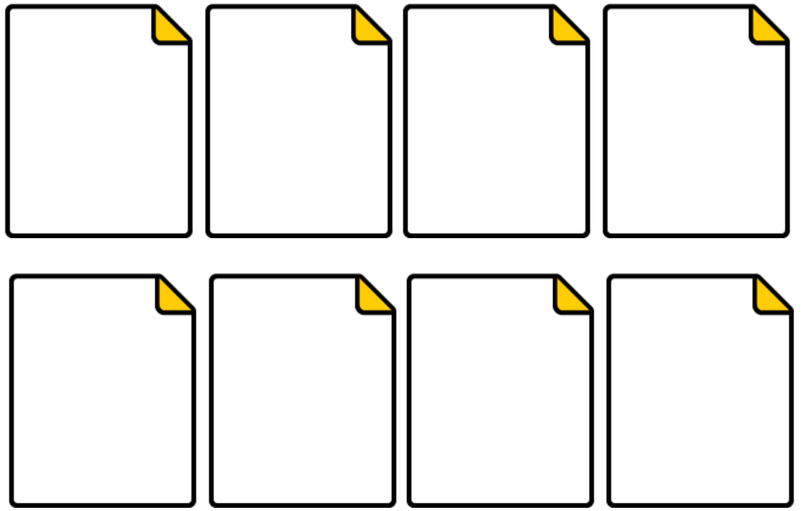
Zhou D. et al. Regularized minimax conditional entropy for crowdsourcing. 2015

Properties of MV

› All workers are treated similarly

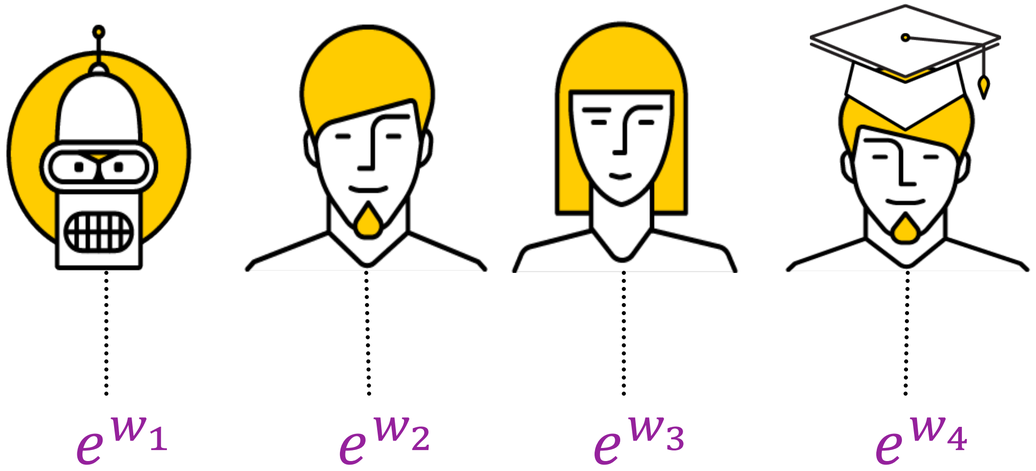


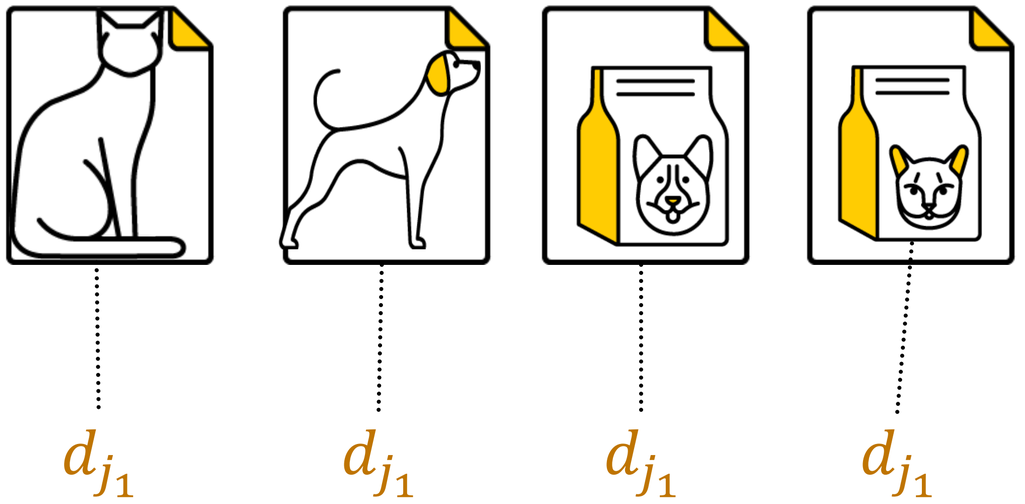
› All objects are treated similarly



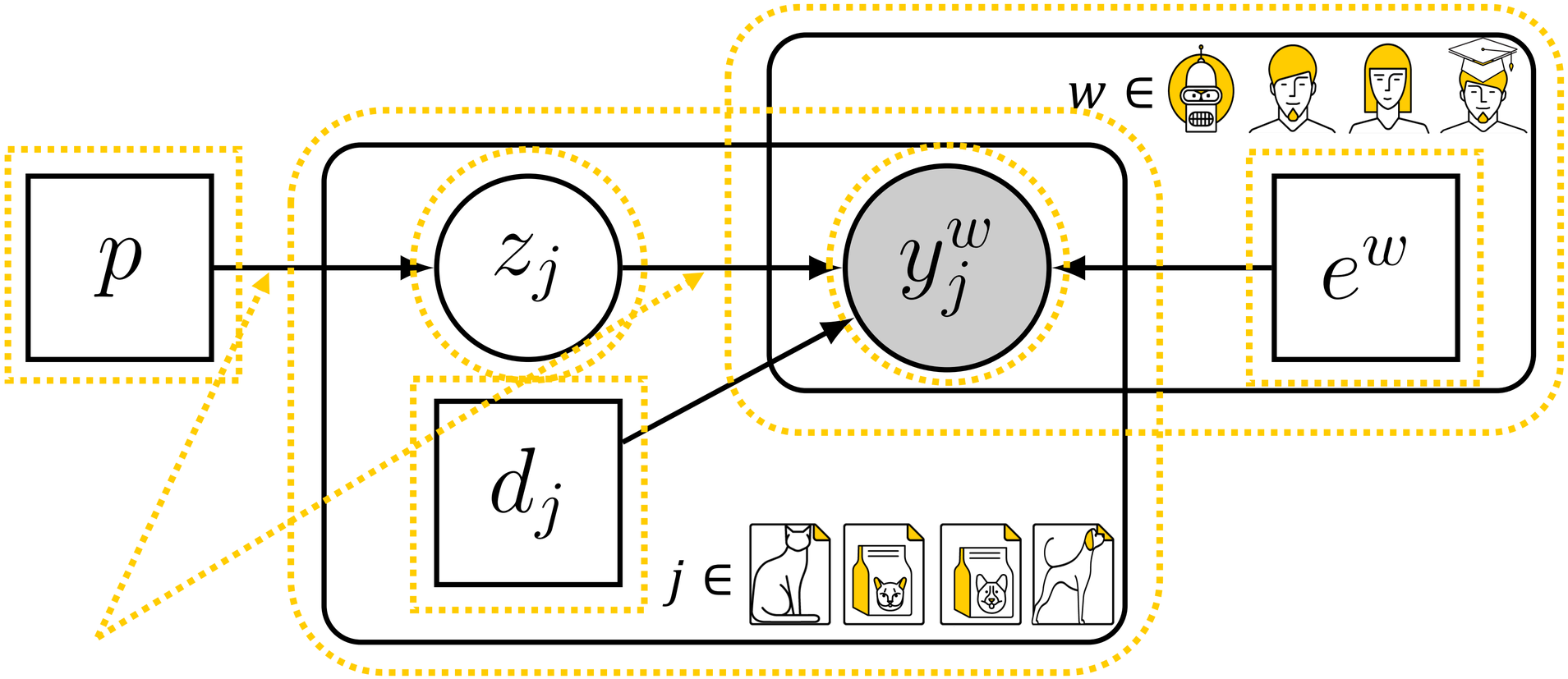
Advanced aggregation: workers and objects

- Parameterize expertise of workers by e^w
- Parameterize difficulty of objects by d_j

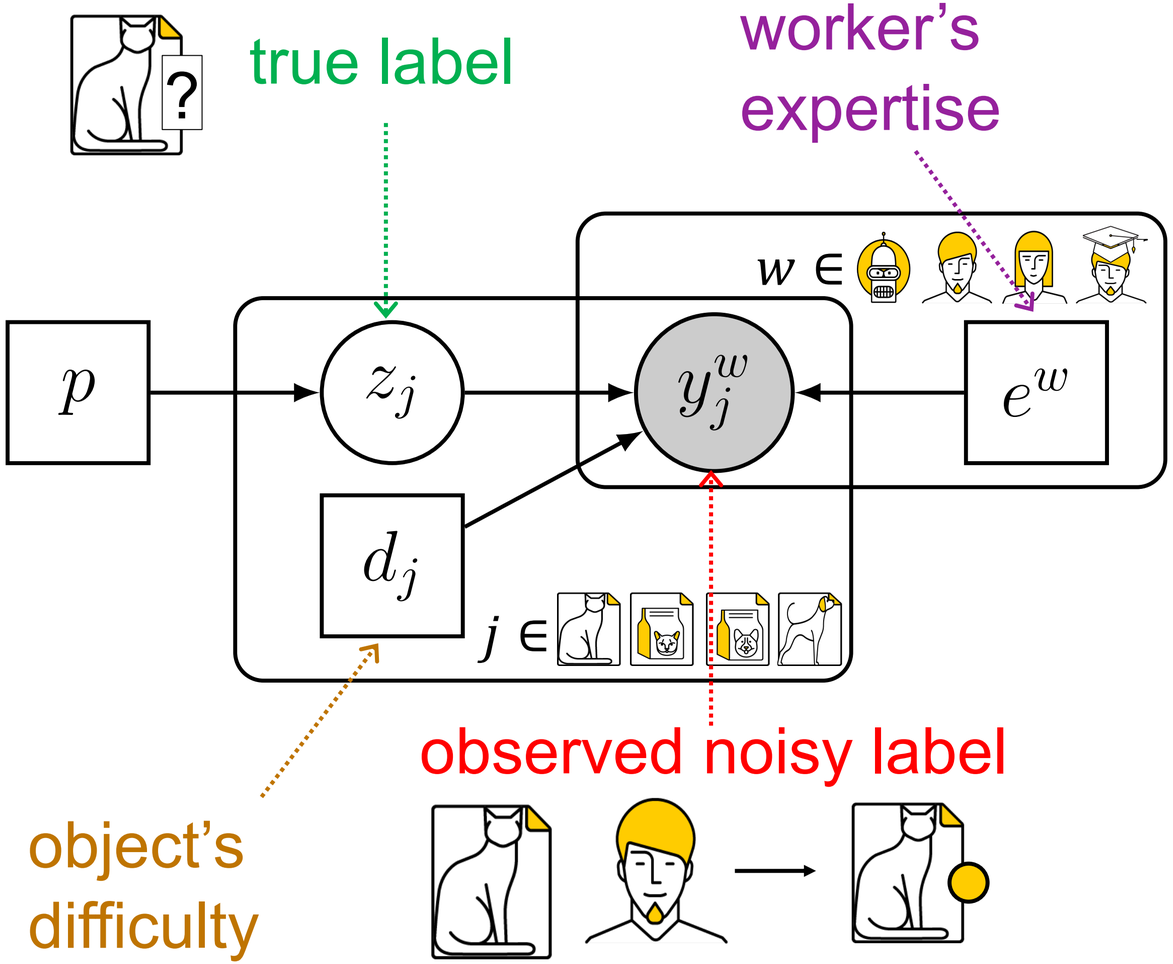




Advanced aggregation: latent label models

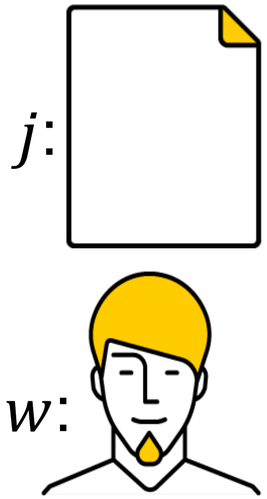


Latent label models: noisy label model



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

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

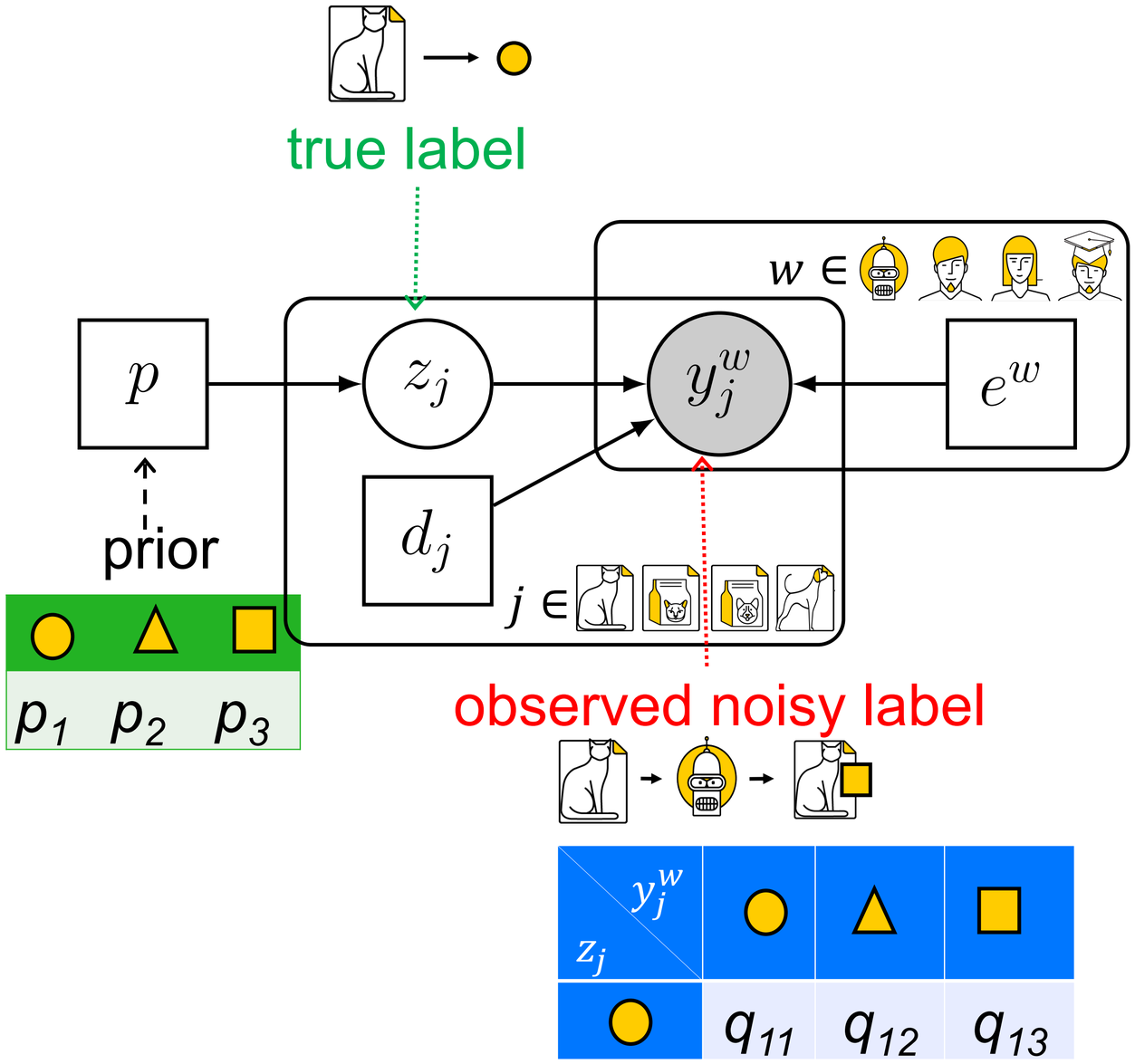


M_j^w:

True \ Noisy	●	▲	■
●	q_{11}	q_{12}	q_{13}
▲	q_{21}	q_{22}	q_{23}
■	q_{31}	q_{32}	q_{33}

$q_{c1} + q_{c2} + q_{c3} = 1$ 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

   $P_Y\left(M_j^w[z_j,\cdot]\right)$

In multiclassification, a standard choice for $P_Z(\cdot)$ and $P_Y(\cdot)$ is a Multinomial distribution $\mathrm{Mult}(\cdot)$

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 $\mathbf{y}$ and $\mathbf{z}$ under the latent label model:

observed noisy label

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

latent true label

latent parameters

likelihood of noisy and true labels for object j

- Estimate parameters and true labels by maximizing $L(\ldots)$

Latent label models: EM algorithm

- Maximization of the expectation of log-likelihood (LL)*

$$\mathbb{E}_{\mathbf{z}} \log \Pr(\mathbf{y}, \mathbf{z}) = \sum_{j \in \mathbb{J}} \sum_{z_j \in \{1,\ldots,K\}} \Pr(z_j | p) \log \prod_{w \in W_j} \Pr(z_j | p) \Pr(y_j^w | z_j, d_j, e^w)$$

- **E-step**: Use Bayes' theorem for posterior distribution of $\hat{\mathbf{z}}$ given p, $\mathbf{d}$, $\mathbf{e}$:

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

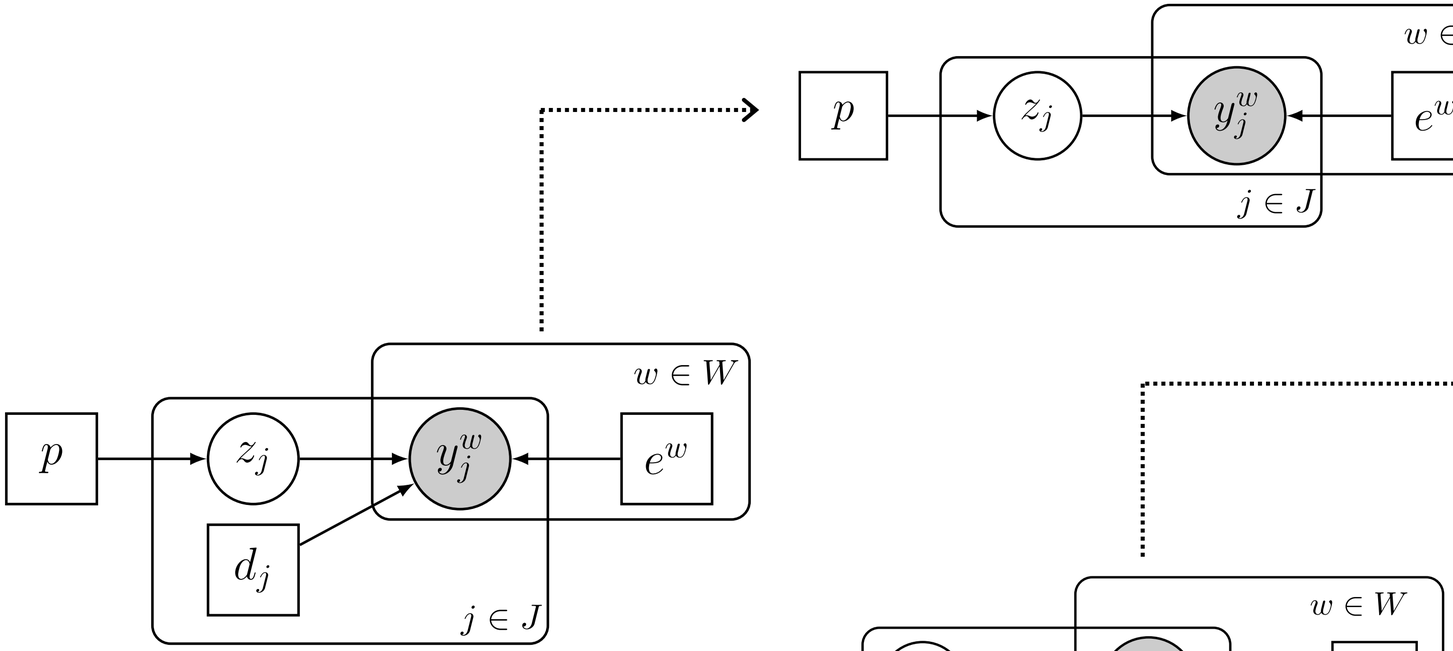
- **M-step**: Maximize the expectation of LL with respect to the posterior distribution of $\hat{\mathbf{z}}$:

$$(p, \mathbf{d}, \mathbf{e}) = \operatorname{argmax} \mathbb{E}_{\hat{\mathbf{z}}} \log \Pr(z_j | p) \prod_{w \in W_j} \Pr(y_j^w | z_j, d_j, e^w)$$

  - Analytical solutions
  - Gradient descent

* it is a lower bound on LL of $\mathbf{y}$ and $\mathbf{z}$

Latent label model (LLM): special cases

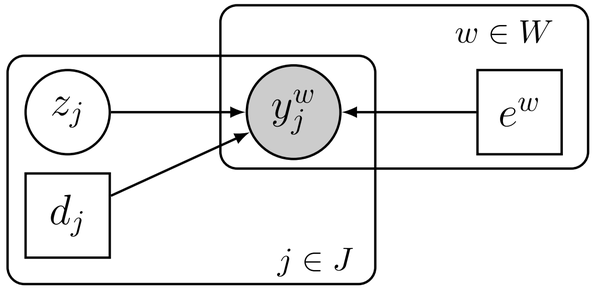


Dawid and Skene model (DS):

- categories are different
- objects are similar
- workers are different

Generative model of labels, abilities, and difficulties (GLAD):

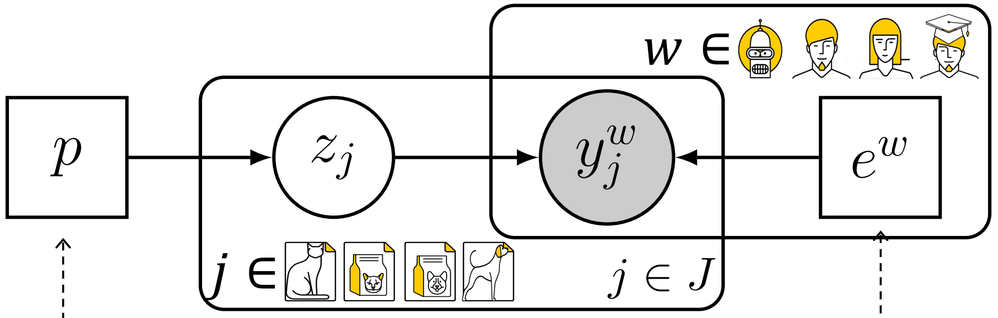
- categories are similar
- objects are different
- workers are different



Minimax conditional entropy model (MMCE):

- categories are different
- objects are different
- workers are different

Dawid and Skene model (DS)[1]



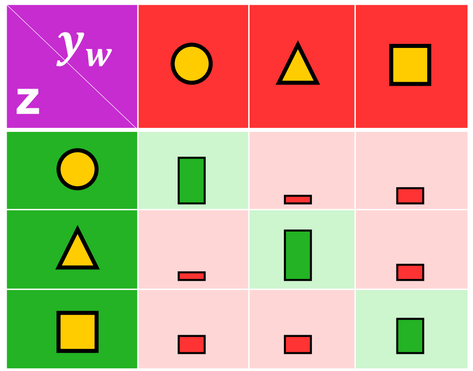
class prior

worker's confusion matrix

LLM with parameters:

- p – vector of length K: $p[i] = \Pr(Z = c)$
- e^w – matrix of size $K \times K$:

$$e^w[c,k] = \Pr(Y^w = k | Z = c)$$



1. Dawid and Skene, "Maximum Likelihood Estimation of Observer Error-Rates Using the *EM* Algorithm",1979

DS: parameters optimization

- **E-step**:

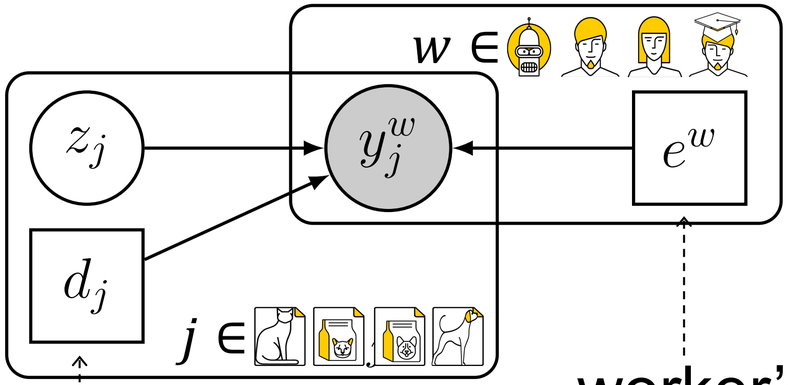
$$\hat{z}_j[c] = \frac{p[c] \prod_{w \in W_j} e^w\left[c, y_j^w\right]}{\sum_k p[k] \prod_{w \in W_j} e^w\left[k, y_j^w\right]}, \qquad c = 1, \ldots, K$$

- **M-step**: Analytical solution

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

$$p[c] = \frac{\sum_{j \in J} \hat{z}_j[c]}{J}, \qquad c = 1, \ldots, K$$

Generative model of Labels, Abilities, and Difficulties (GLAD)[2]



worker's ability

object's inverse difficulty

LLM with parameters:

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

$$\Pr\left(Y_j^w = k | Z_j = c\right) = \begin{cases} a(w,j), & c = k \\ \dfrac{1 - a(w,j)}{K-1}, & c \neq k \end{cases}$$

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

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

GLAD: parameters optimization

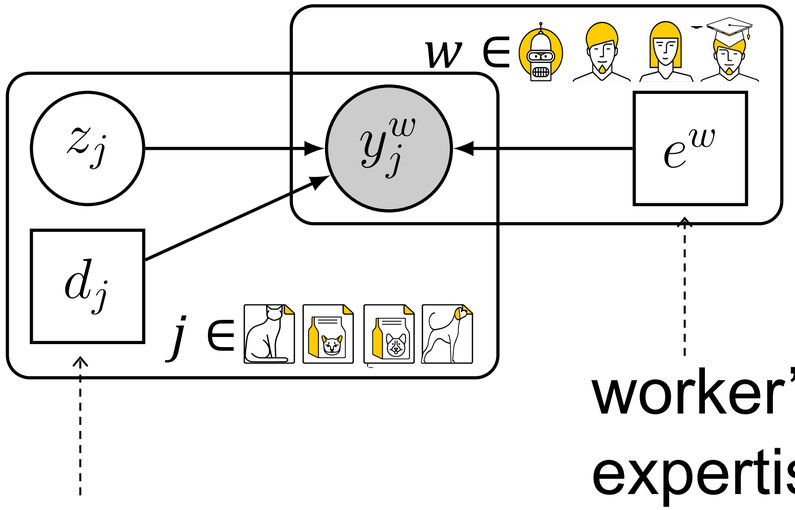
- Let $a(w,j) = \frac{1}{1+\exp(-e^w d_j)}$ and $\mathrm{P}(z_j)$ be a predefined prior (e.g., $\mathrm{P}(z_j) = {}^1/_{\boldsymbol{K}}$)
- **E-step**:

$$\widehat{z_j}\,[c] \propto \mathrm{P}(Z_j = c) \prod_{w \in W_j} a(w,j)^{\delta\left(y_j^w = c\right)} \left(\frac{1 - a(w,j)}{K-1}\right)^{\delta\left(y_j^w \neq c\right)}, \quad c = 1, \ldots, K$$

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

$$(\mathbf{d}^{\mathrm{t}}, \mathbf{e}^{\mathrm{t}}) = \operatorname{argmax} \sum_{j \in J} \left[\mathbb{E}_{\widehat{z_j}} \log \mathrm{P}(z_j) + \sum_{w \in W_j} \mathbb{E}_{\widehat{z_j}} \log \Pr\left(y_j^w | z_j\right) \right]$$

MiniMax Conditional Entropy model (MMCE)[3]



object's confusability matrix

worker's expertise matrix

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

$$\min_Q \max_P - \sum_{\substack{j \in J \\ c \in \{1,\dots,K\}}} Q(Z_j = c) \sum_{\substack{w \in W \\ k \in \{1,\dots,K\}}} P(Y_j^w = k | Z_j = c) \log P(Y_j^w = k | Z_j = c)$$

LLM with parameters:

› d_j – matrix of size $K \times K$

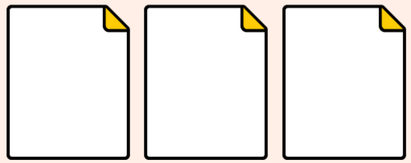
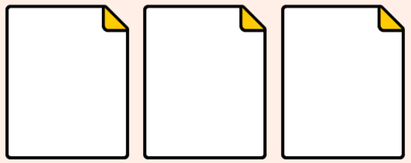
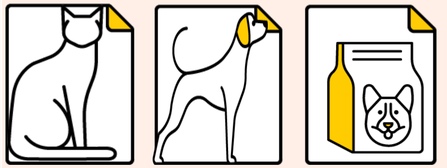
› e^w – matrix of size $K \times K$

› Noisy label model:

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

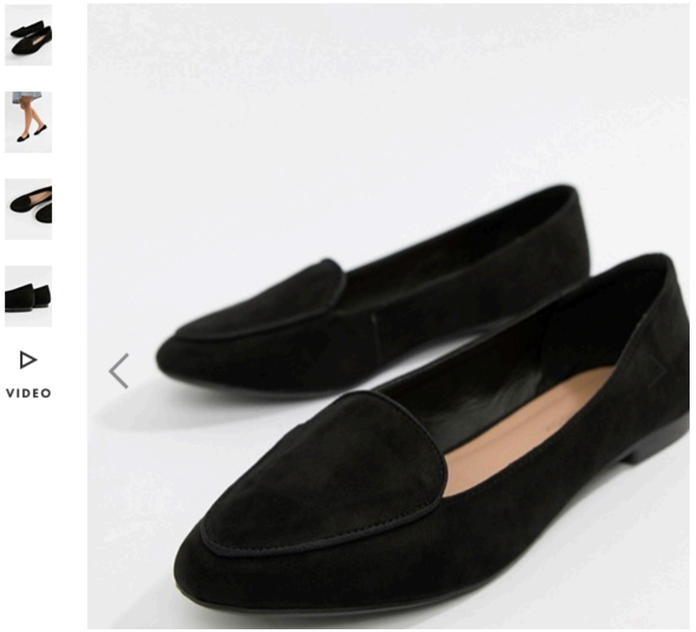
3. Zhou et al., Learning from the Wisdom of Crowds by Minimax Entropy, 2012

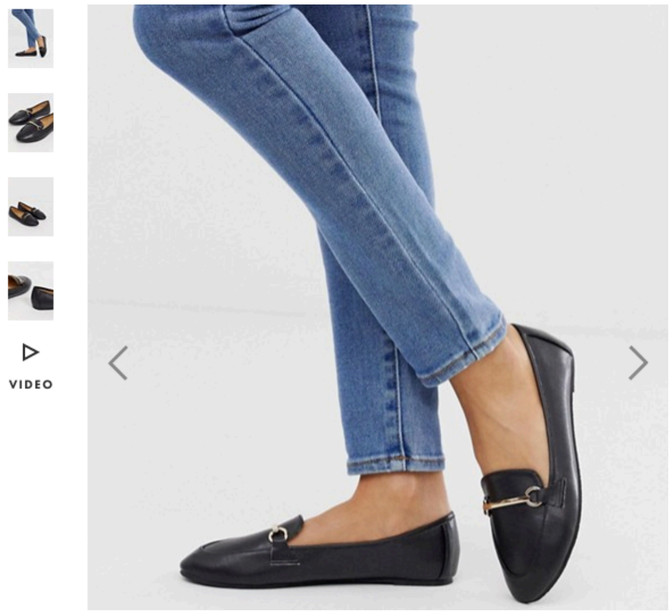
Summary of aggregation methods

	MV	DS	GLAD	MMCE
Categories (K)				
Objects (J)				
Workers (W)				
Number of parameters	0	$WK^2 + K$	$W + J$	$(W + J)K^2$

Pairwise comparisons

Project 4: Compare items





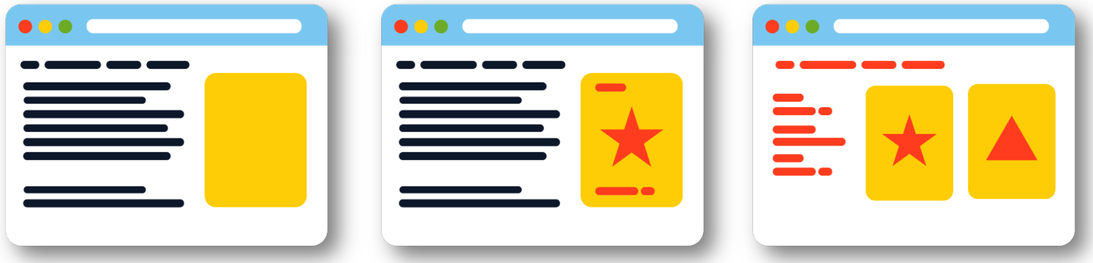
Which shoes look more similar to the one in the picture?

Left

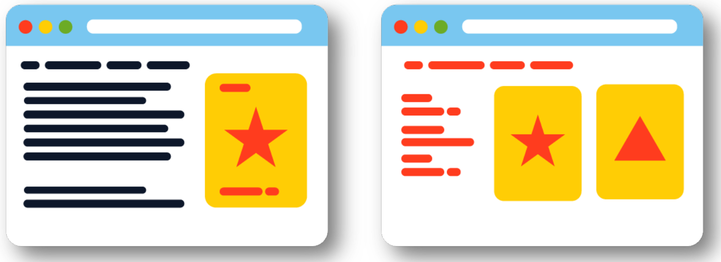
Right

Notation

- Answers: **Left** or **Right**
- Items $d_j \in \{1, \ldots, N\}$. E.g.:

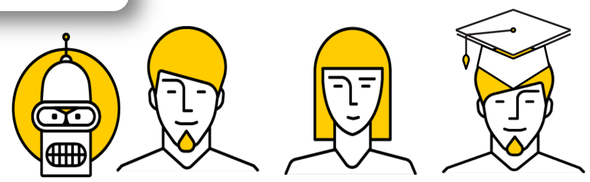


- Tasks:



Choose a better item:
Left
Right

- Workers: $w \in \{1, \ldots, W\}$. E.g.:

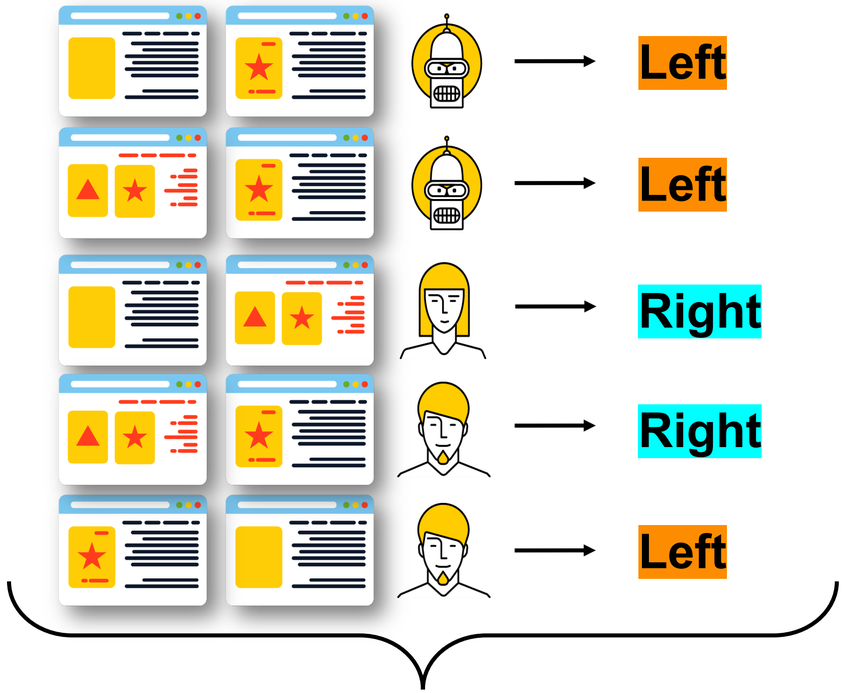


Formalization

Ranking from pairwise comparisons:

› Given pairwise comparisons for items in D:

$$P = \{(w_k, d_i, d_j) : i \succ_k j\}$$

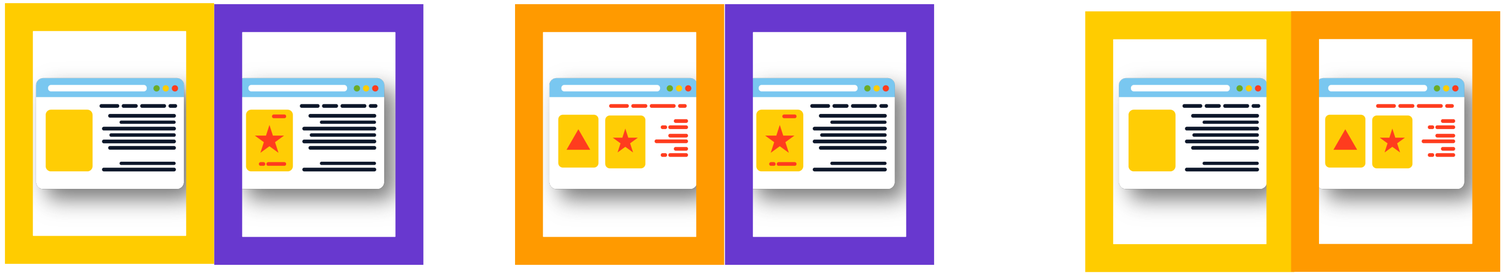


› Obtain **a ranking** π over items $D \to \{1, \ldots, N\}$ based on answers in P

Difference from multiclassification

- The latent label assumption is not satisfied when comparing complex items

- Different tasks may contain common items



Bradley and Terry model (BT)

› Assume that each item $d_i \in D$ has a latent “quality” score $s_i \in \mathbb{R}$

d_i  $\longleftrightarrow$ s_i

› The probability that $d_i \in D$ will be preferred in a comparison over $d_j \in D$

$$\Pr(i \succ j) = f\left(s_i - s_j\right),$$

where $f(x) = {}^{1}/_{1+e^{-x}}$.

› The model assumes that all workers are equally good and truthful

Bradley, R. A. and Terry, M. E. "Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons". 1952

NoisyBT model: parameterization of workers

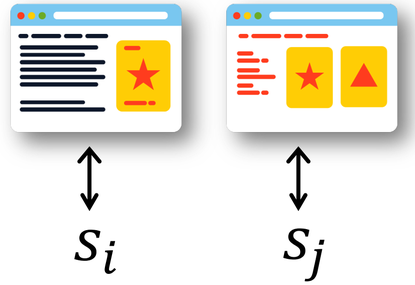
w_k ⟷ "reliability" γ_k and "bias" q_k

- Probability that w_k reads a task is

Logistic function

$$\Pr(w_k \text{ reads a task}) = f(\gamma_k)$$

  - If w_k reads the task, she answers according to scores:



$$(f(s_i - s_j), f(s_j - s_i))$$

Probability to choose **Left** if compares items

  - If w_k does not read the task, she answers according to her bias:

$$(f(q_k), f(-q_k))$$

Probability to choose **Left** if answers randomly

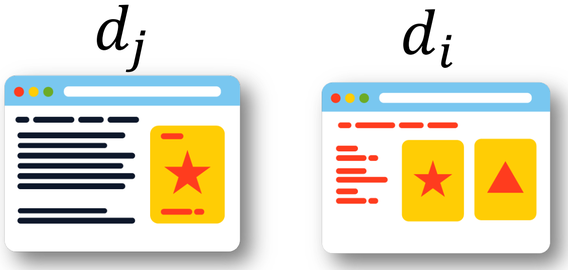
NoisyBT: likelihood of workers' answers

The likelihood of $i \succ_k j$ is

$$\Pr(i \succ_k j) = \underbrace{f(\gamma_k) f(s_i - s_j)}_{\text{Truthful answer}} + \underbrace{\left(1 - f(\gamma_k)\right) f\left((-1)^{(1 - \mathbb{I}(d_i \text{ was left}))} q_k\right)}_{\text{Random answer}},$$

where $\mathbb{I}(d_i \text{ was left})$ is the indicator for the order of d_i and d_j

$\mathbb{I}(d_i \text{ was left}) = 1$



$\mathbb{I}(d_i \text{ was left}) = 0$

NoisyBT: parameters optimization

Likelihood of observed comparisons:

$$T(s, q, \gamma) = \sum_{(w_k, d_i, d_j) \in P} \log \Pr(i \succ_k j) =$$

$$\sum_{(w_k, d_i, d_j) \in P} \log\left[f(\gamma_k) f(s_i - s_j) + \left(1 - f(\gamma_k)\right) g(q_k)\right]$$

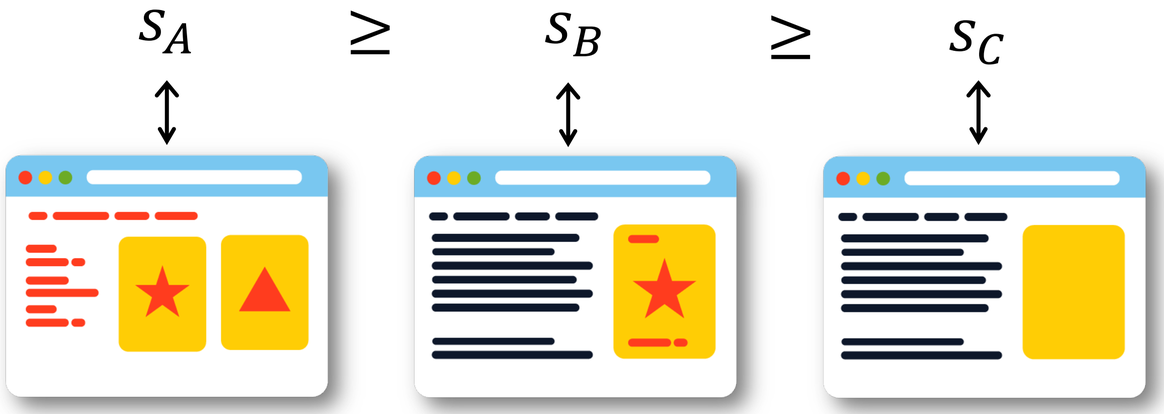
- $\{s_i\}_{i=1,\dots,N}$ and $\{\gamma_k, q_k\}_{k=1,\dots,W}$ are inferred by maximizing the log-likelihood:

$$T(s, q, \gamma) \rightarrow \max_{\{s_i, \gamma_k, q_k\}}$$

- To obtain a **ranking** π over items, **sort** items according to their **scores**

Summary about pairwise comparisons

› Latent scores models for ranking from pairwise comparisons:



› To reduce bias from unreliable answers parameterize workers

⟷ “reliability” γ_k and “bias” q_k

Yandex

Thank you!
Questions?

Valentina Fedorova

Research analysts

valya17@yandex-team.ru

https://research.yandex.com/tutorials/crowd/wsdm-2020