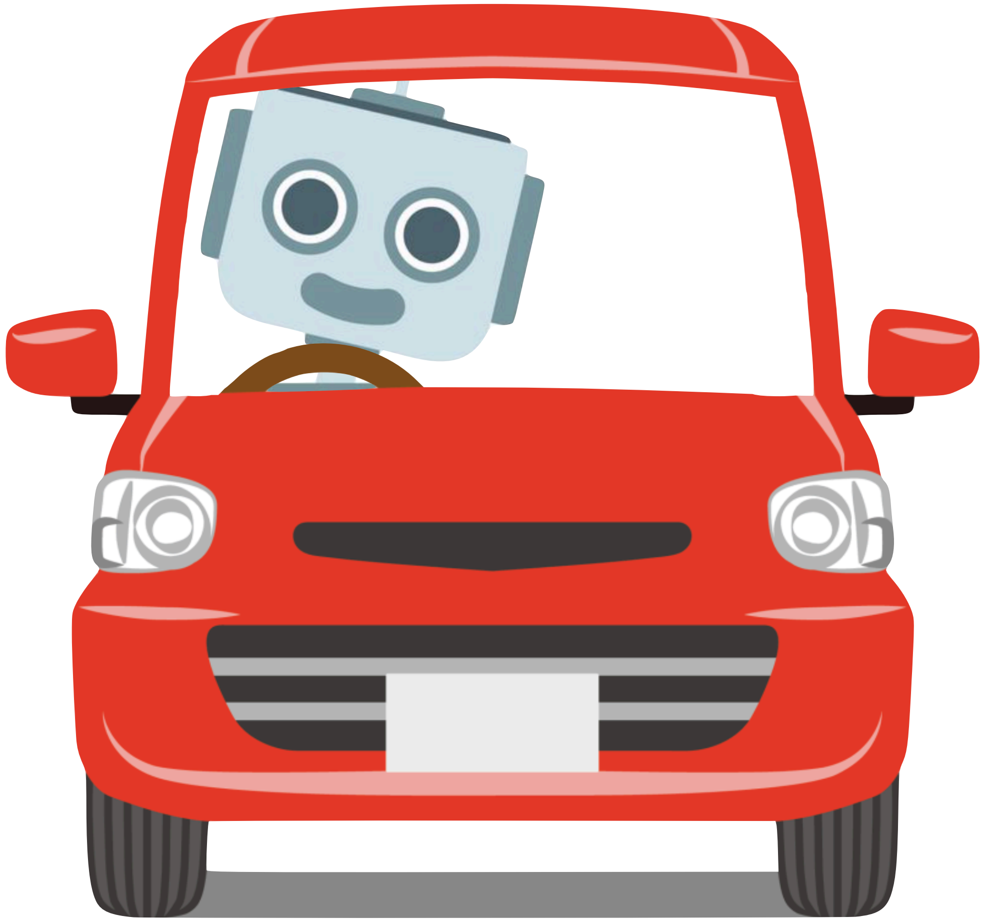
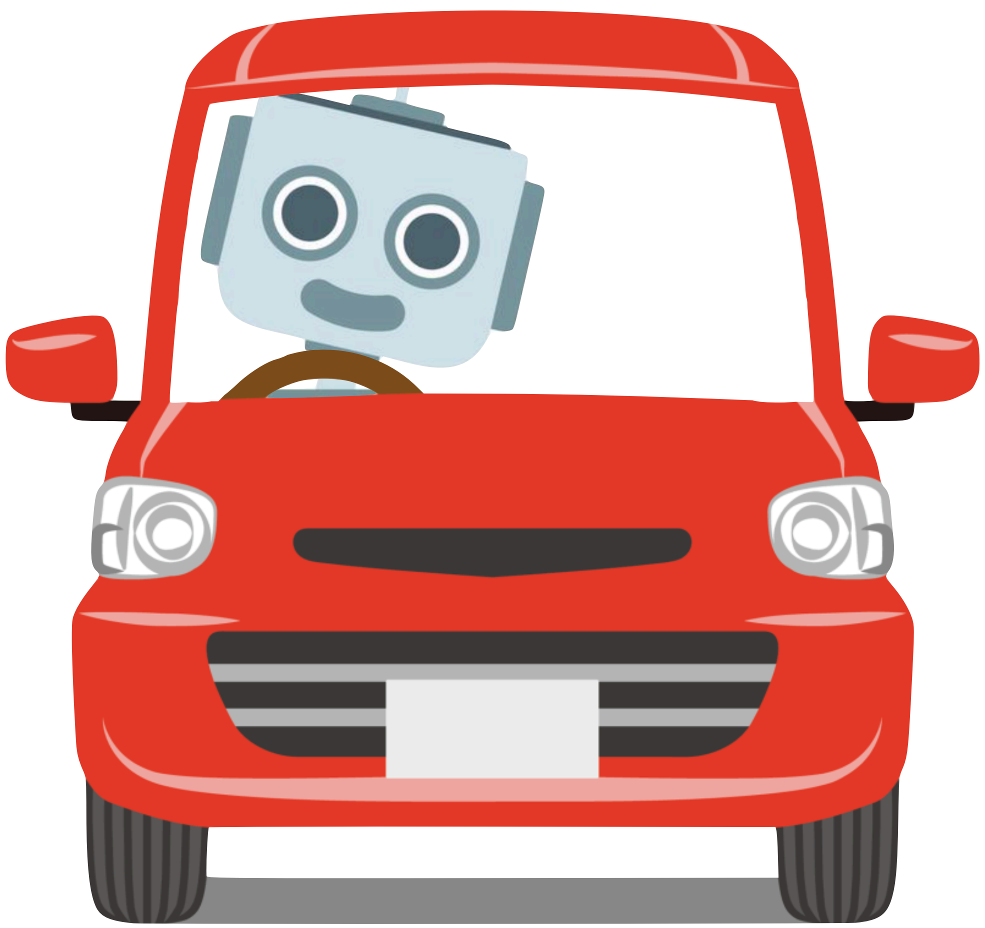
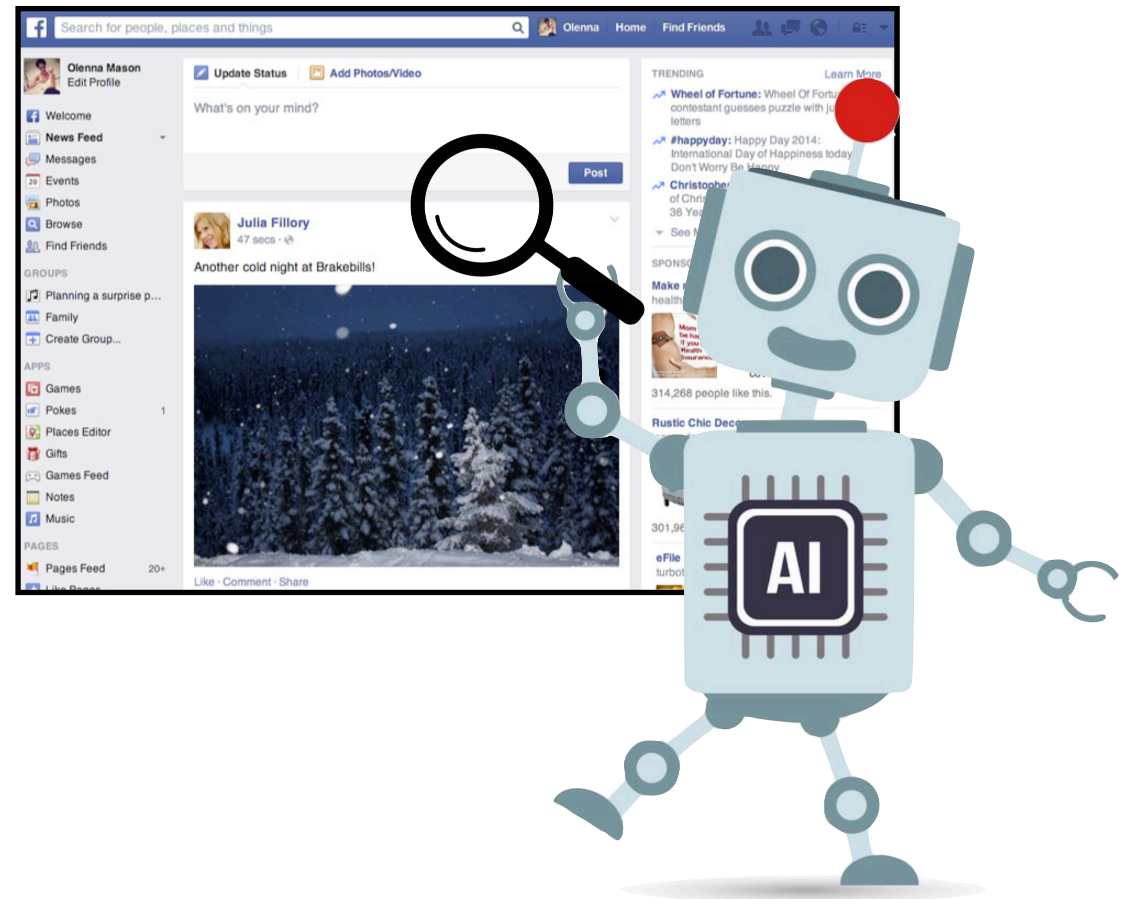
Learning to Defer to One, Multiple, or a Population of Expert(s)

Eric Nalisnick

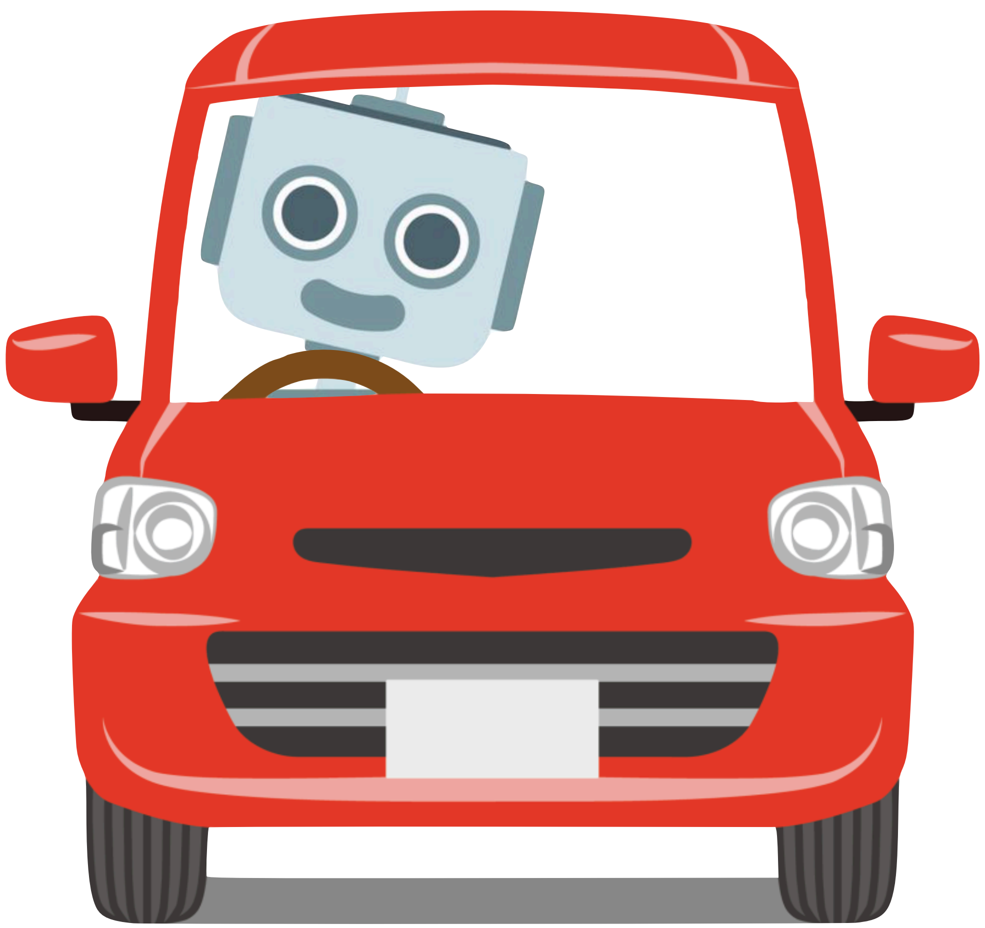
Johns Hopkins University

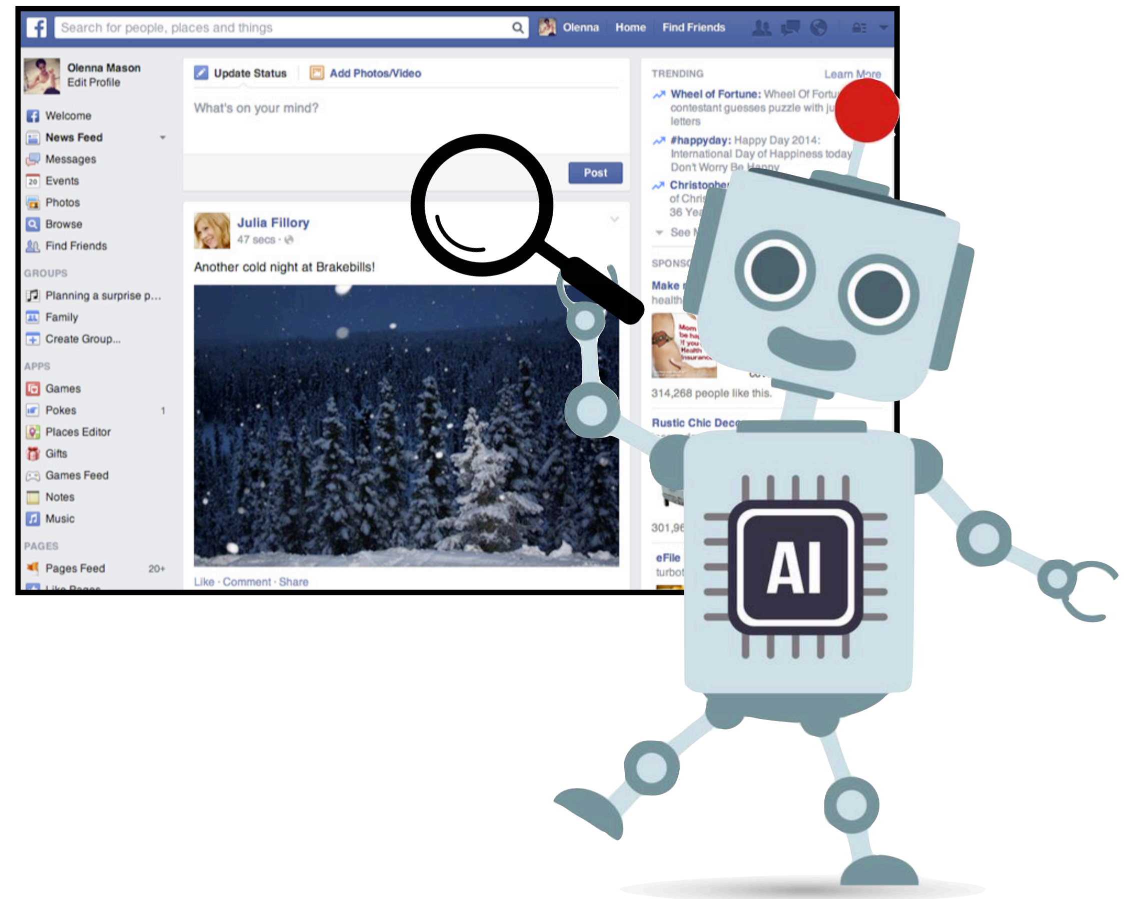


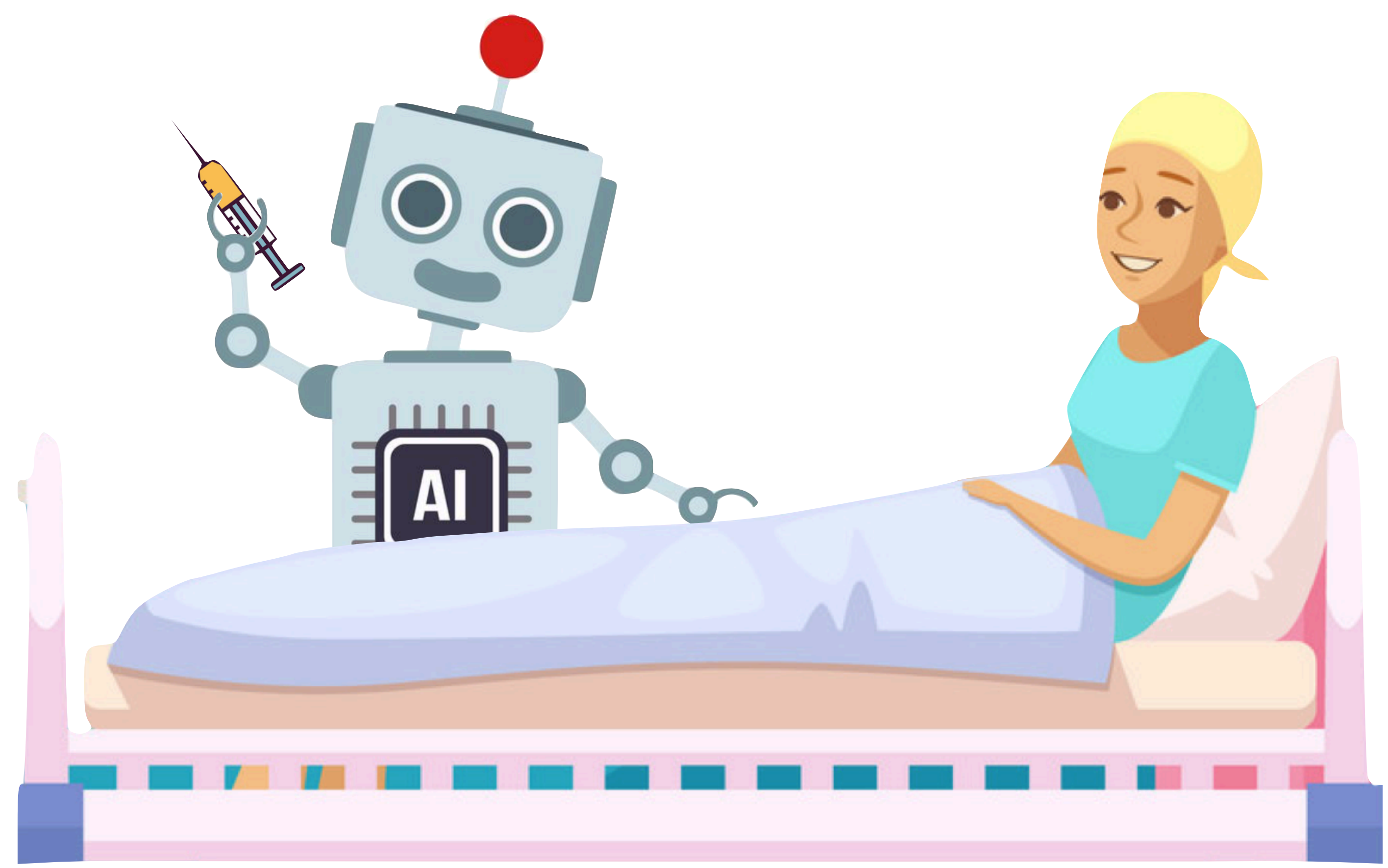


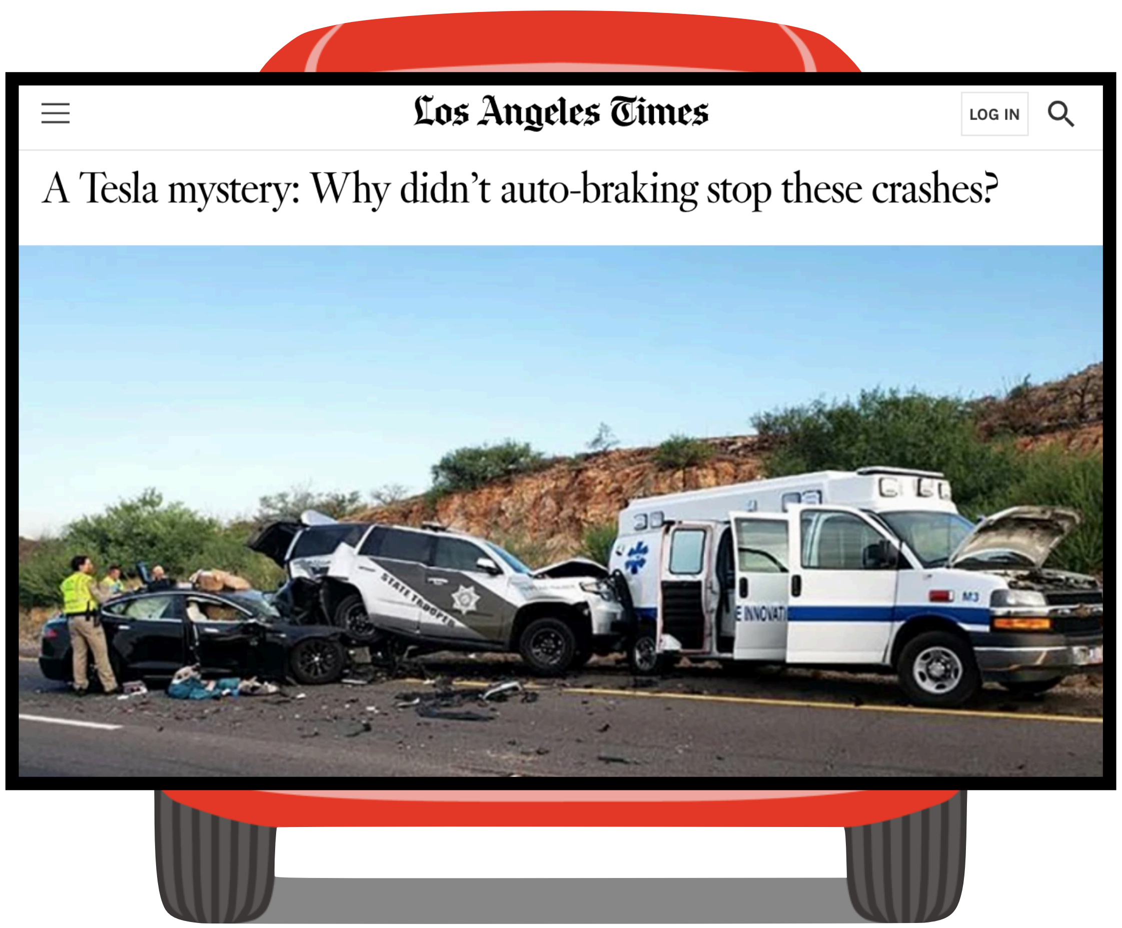


Search for people, places and things
Olenna
Home
Find Friends
Olenna Mason
Edit Profile
Welcome
News Feed
Messages
Events
Photos
Browse
Find Friends
GROUPS
Planning a surprise p...
Family
Create Group...
APPS
Games
Pokes
Places Editor
Gifts
Games Feed
Notes
Music
PAGES
Pages Feed
Update Status
Add Photos/Video
What's on your mind?
Post
Julia Fillory
47 secs
Another cold night at Brakebills!
Like · Comment · Share
TRENDING
Learn More
314,268 people like this.
AI

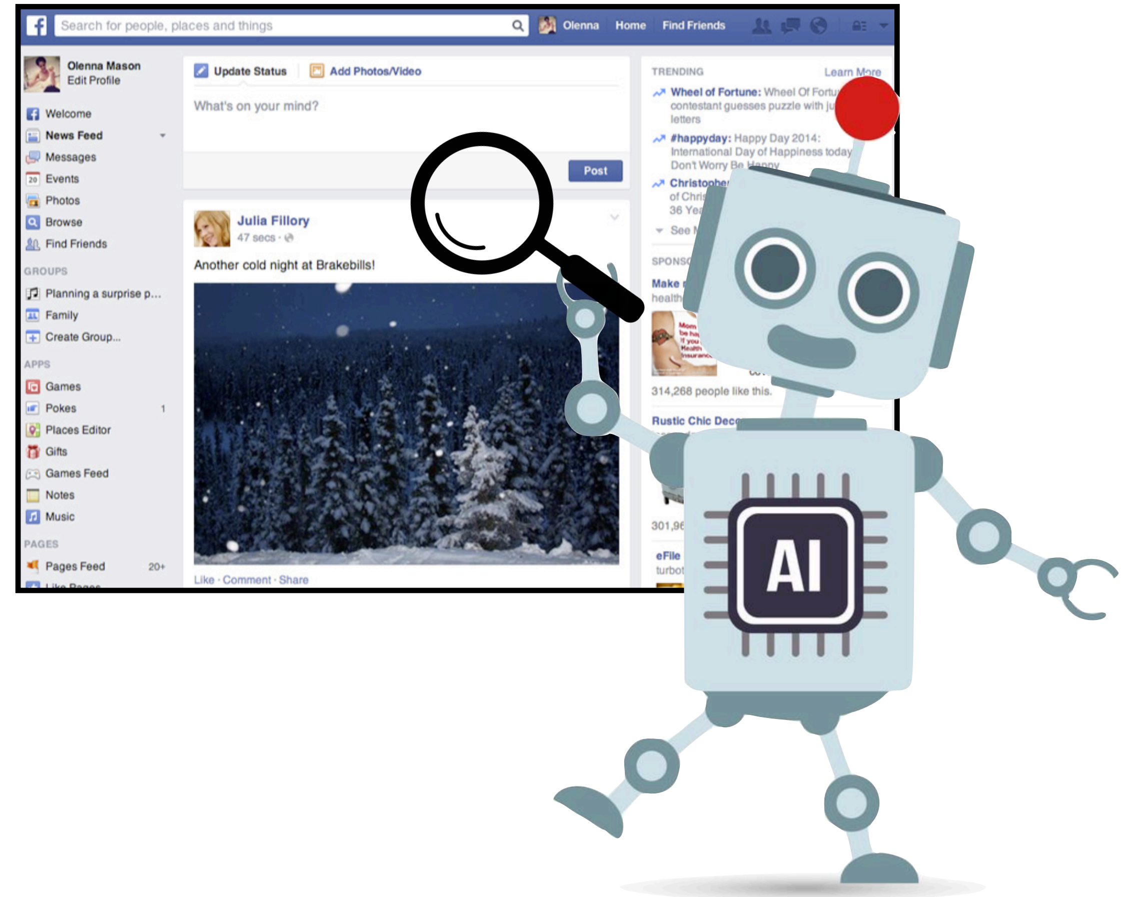



Search for people, places and things
Olenna
Home
Find Friends
Olenna Mason
Edit Profile
Welcome
News Feed
Messages
Events
Photos
Browse
Find Friends
GROUPS
Planning a surprise p...
Family
Create Group...
APPS
Games
Pokes
Places Editor
Gifts
Games Feed
Notes
Music
PAGES
Pages Feed
Update Status
Add Photos/Video
What's on your mind?
Post
Julia Fillory
Another cold night at Brakebills!
Like · Comment · Share
TRENDING
Learn More
314,268 people like this.
AI

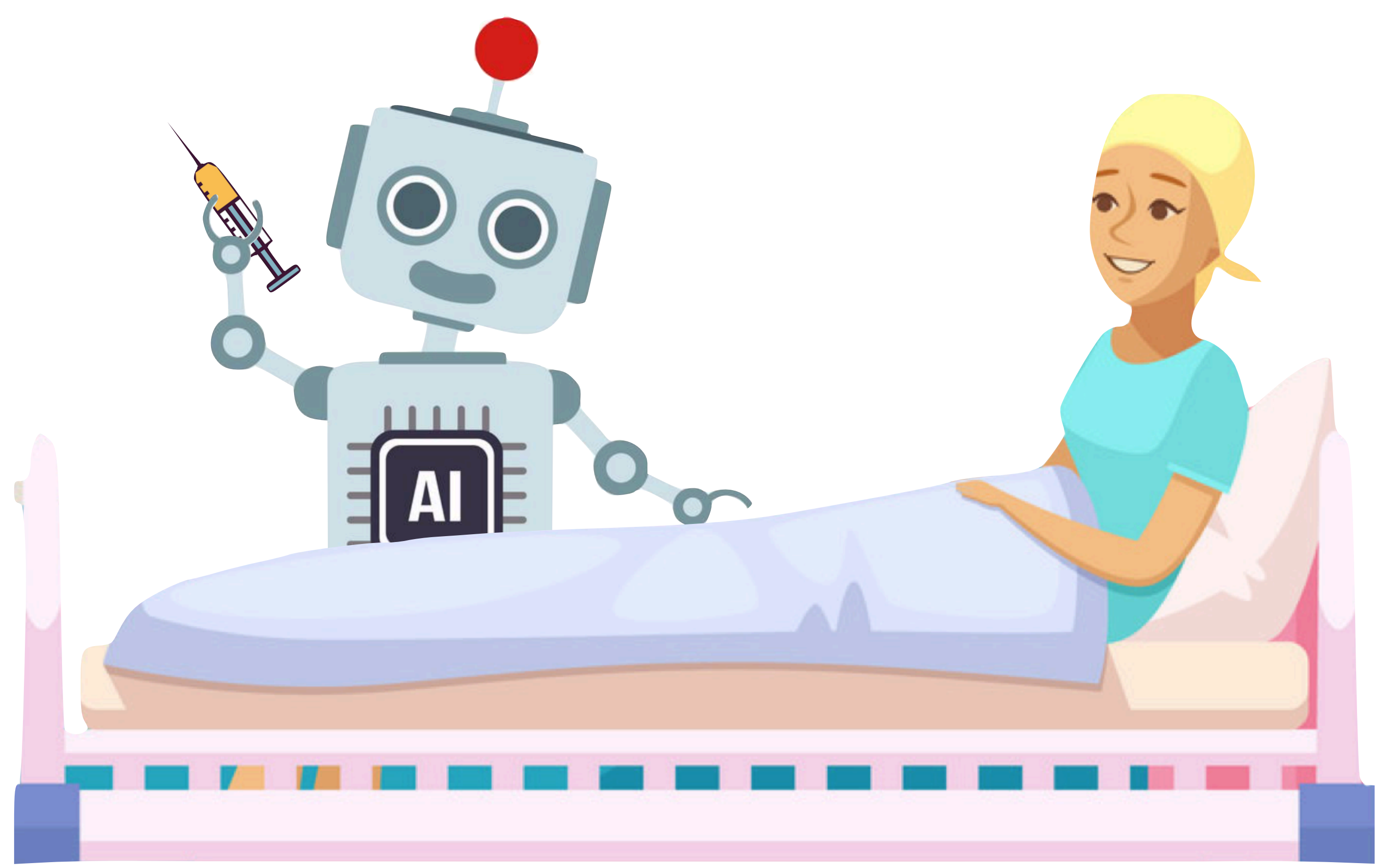

AI



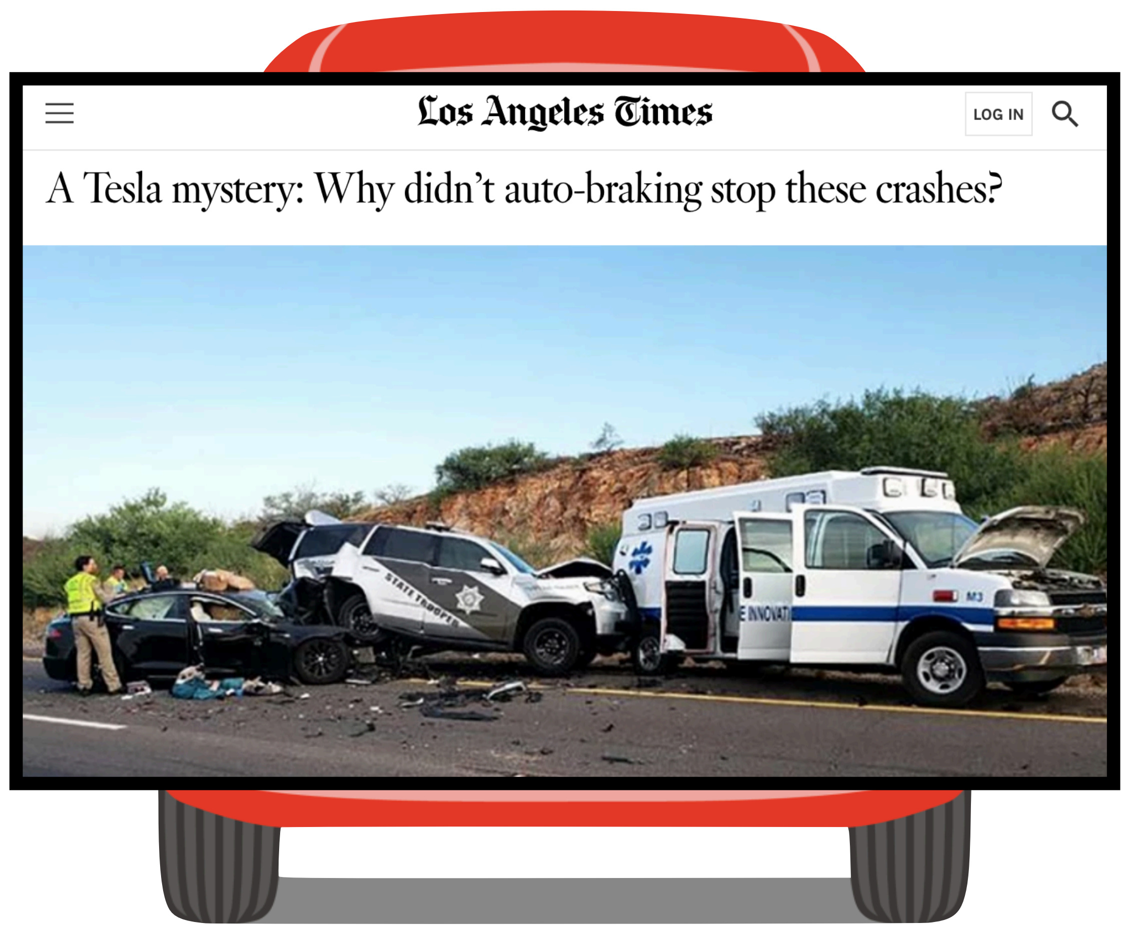
Los Angeles Times
LOG IN
A Tesla mystery: Why didn't auto-braking stop these crashes?



Search for people, places and things
Olenna
Home
Find Friends
Olenna Mason
Edit Profile
Welcome
News Feed
Messages
Events
Photos
Browse
Find Friends
GROUPS
Planning a surprise p...
Family
Create Group...
APPS
Games
Pokes
Places Editor
Gifts
Games Feed
Notes
Music
PAGES
Pages Feed
Update Status
Add Photos/Video
What's on your mind?
Post
Julia Fillory
Another cold night at Brakebills!
Like · Comment · Share
TRENDING
Learn More
314,268 people like this.
AI

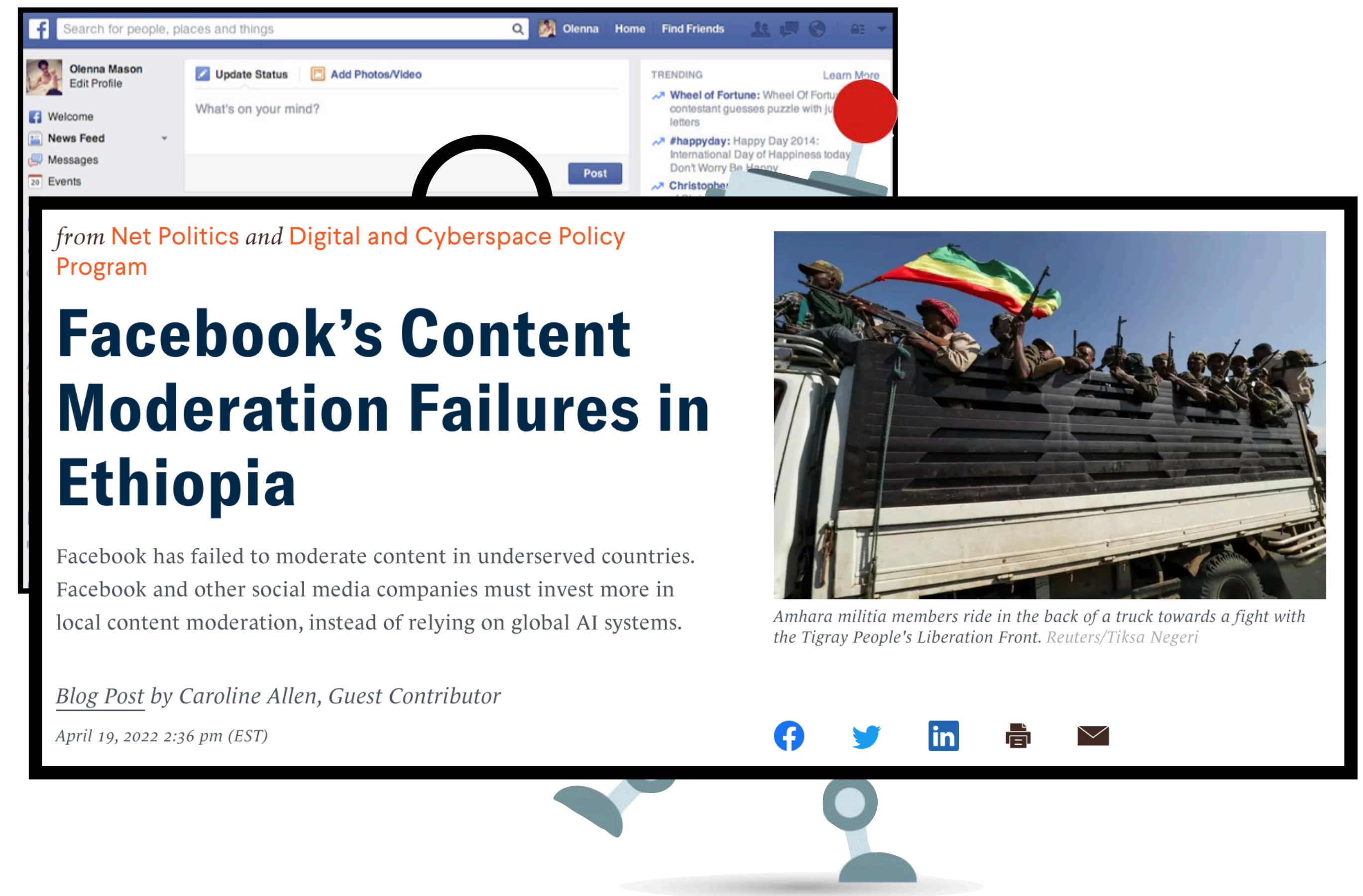


AI


Los Angeles Times

LOG IN

A Tesla mystery: Why didn't auto-braking stop these crashes?


from Net Politics *and* Digital and Cyberspace Policy Program

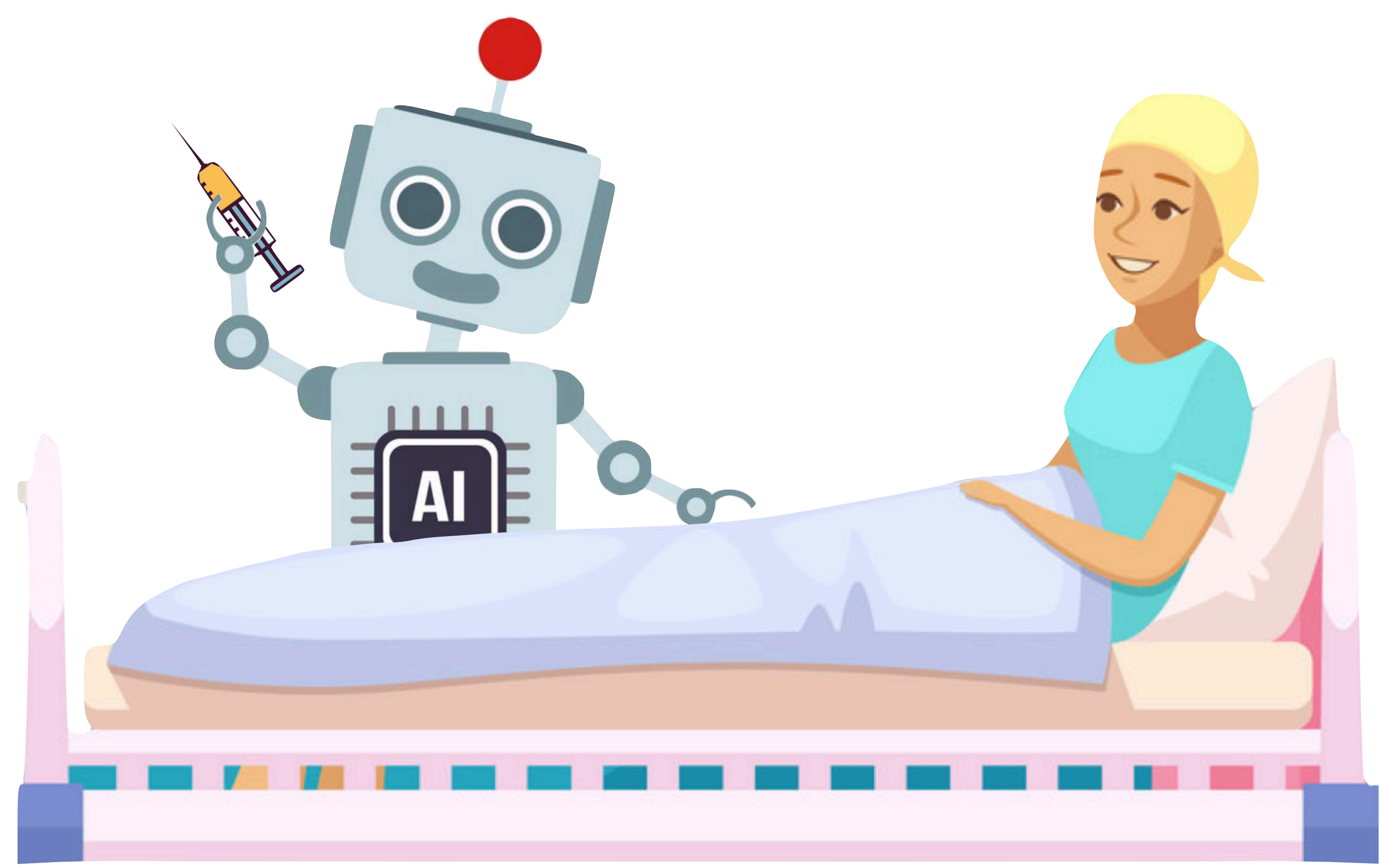
Facebook's Content Moderation Failures in Ethiopia

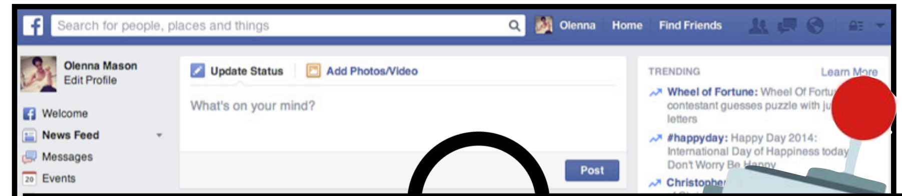
Facebook has failed to moderate content in underserved countries. Facebook and other social media companies must invest more in local content moderation, instead of relying on global AI systems.

Amhara militia members ride in the back of a truck towards a fight with the Tigray People's Liberation Front. Reuters/Tiksa Negeri

Blog Post by Caroline Allen, Guest Contributor

April 19, 2022 2:36 pm (EST)





from Net Politics *and* Digital and Cyberspace Policy Program

Facebook's Content Moderation Failures in Ethiopia

Facebook has failed to moderate content in underserved countries. Facebook and other social media companies must invest more in local content moderation, instead of relying on global AI systems.

Amhara militia members ride in the back of a truck towards a fight with the Tigray People's Liberation Front. Reuters/Tiksa Negeri

Blog Post by Caroline Allen, Guest Contributor

April 19, 2022 2:36 pm (EST)

Los Angeles Times LOG IN

A Tesla mystery: Why didn't auto-braking stop these crashes?

ARTIFICIAL INTELLIGENCE

Google's medical AI was super accurate in a lab. Real life was a different story.

If AI is really going to ma
works when real humar

By Will Douglas Heaven

Medscape Tuesday, December 13, 2022

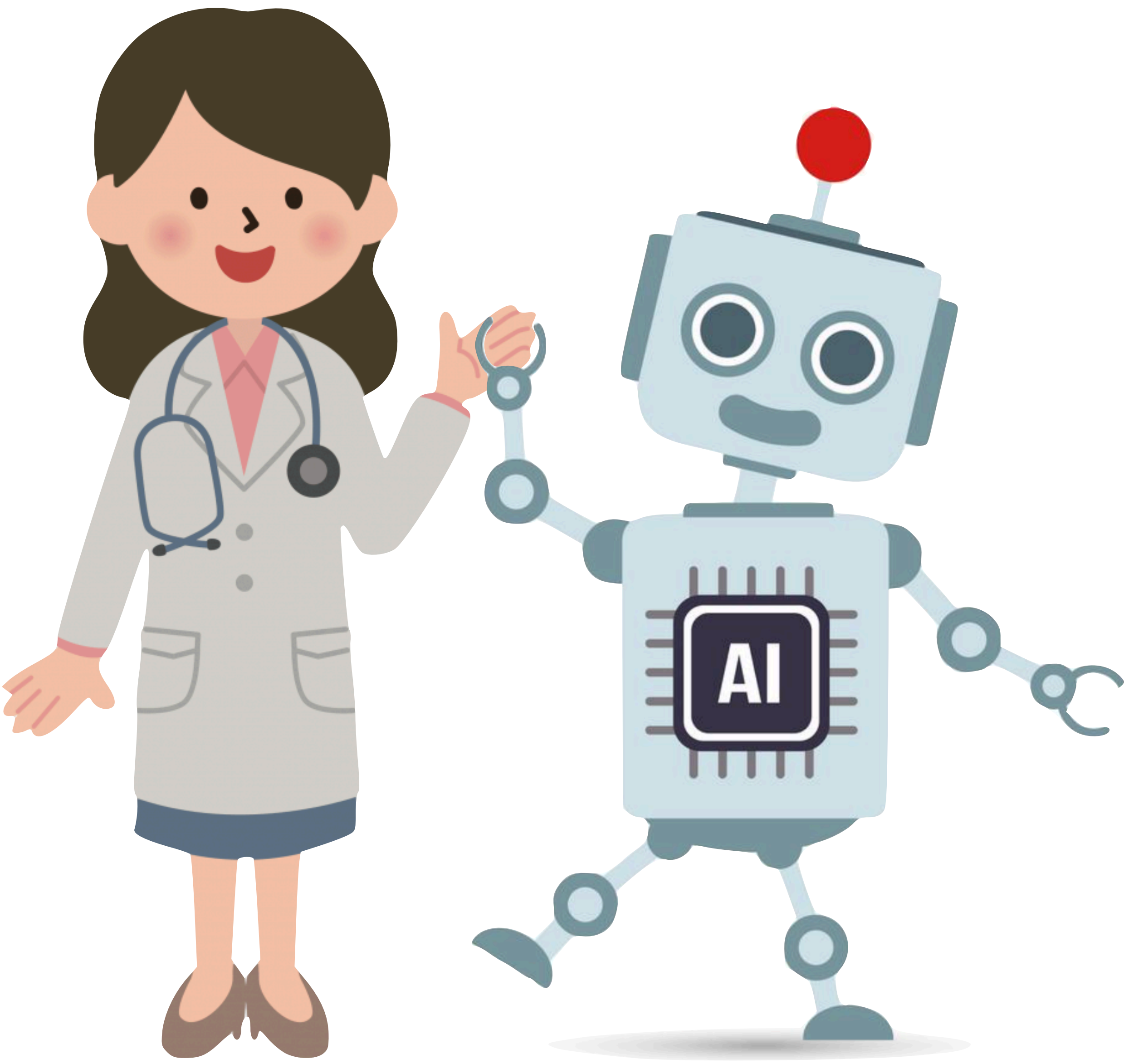
NEWS & PERSPECTIVE | DRUGS & DISEASES | CME & EDUCATION | ACADEMY | VIDEO | DECISION POINT

News > Medscape Medical News > Conference News > CHEST 2022

Sepsis Predictor Tool Falls Short in Emergency Setting

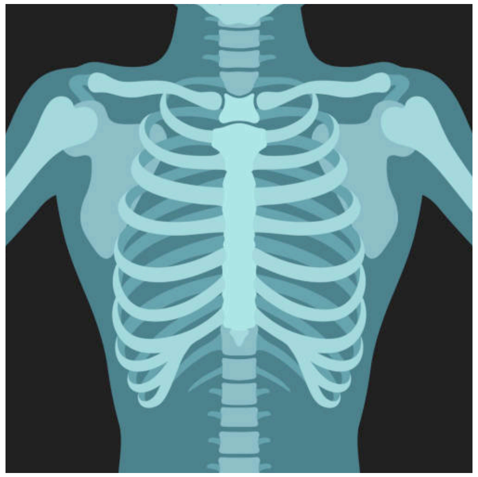
Heidi Splete

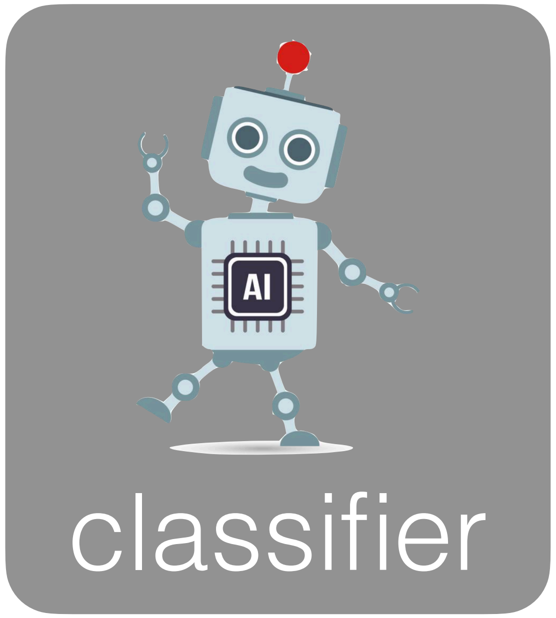
October 17, 2022



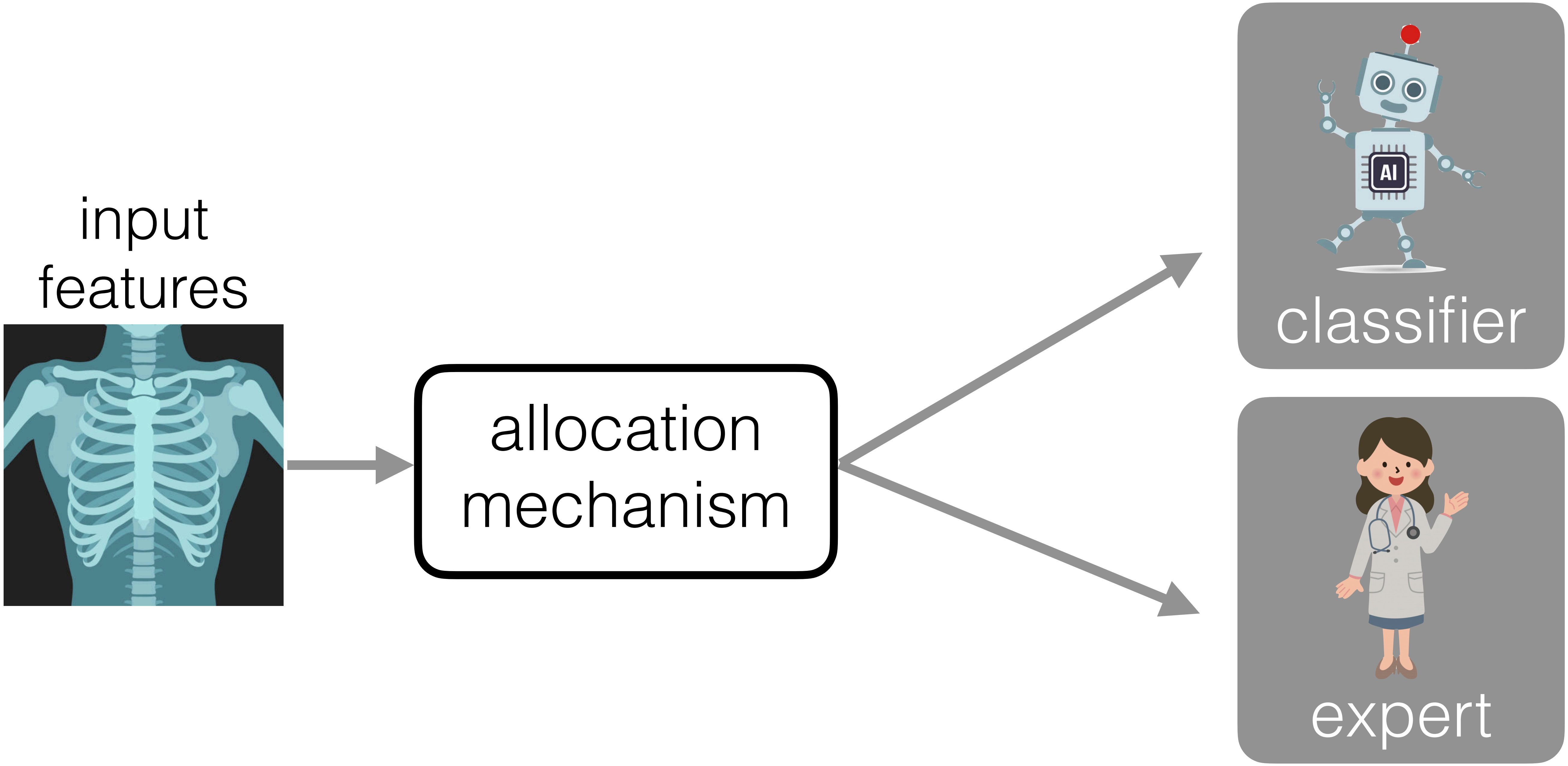
human-AI collaboration

input features

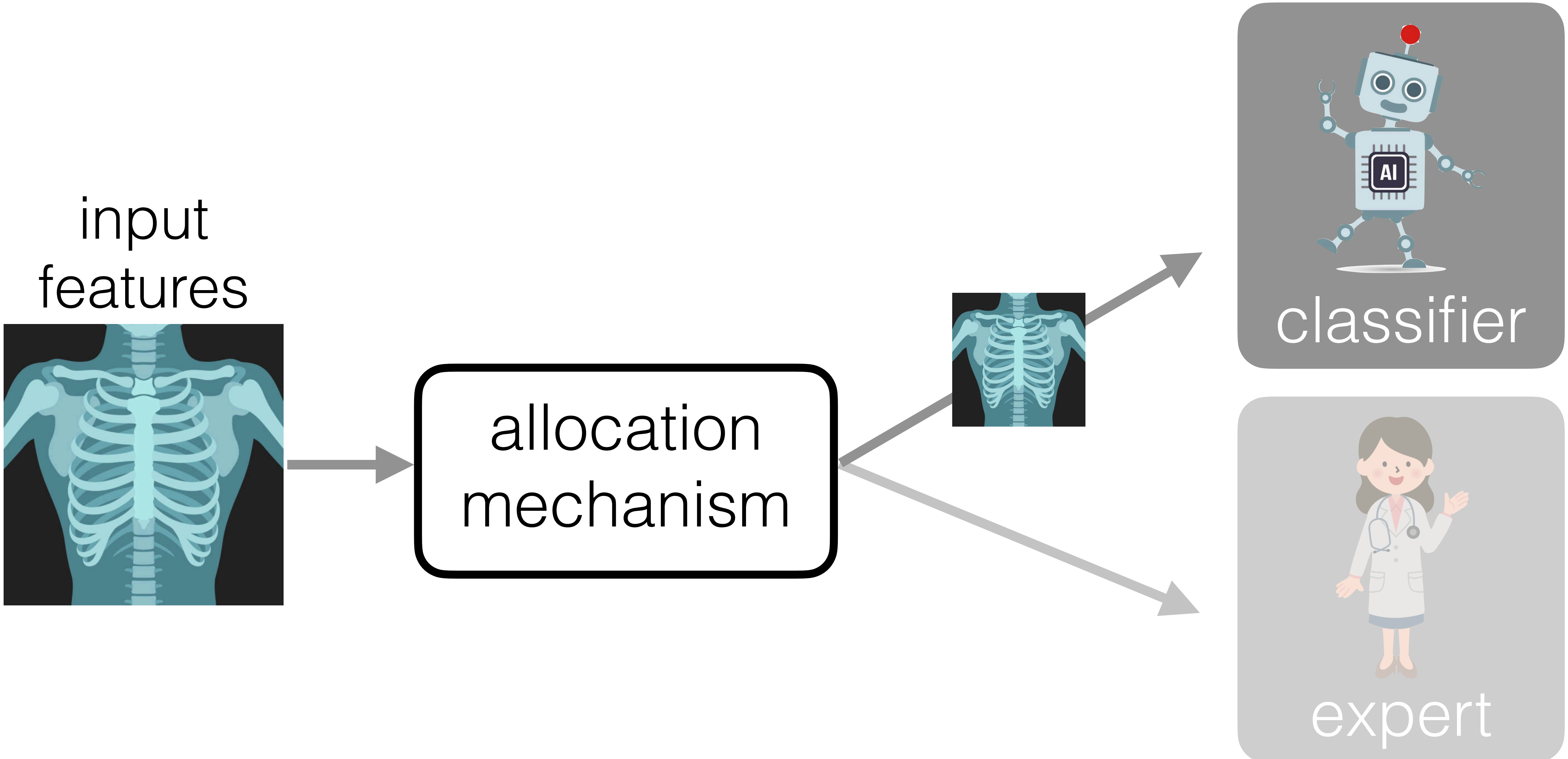




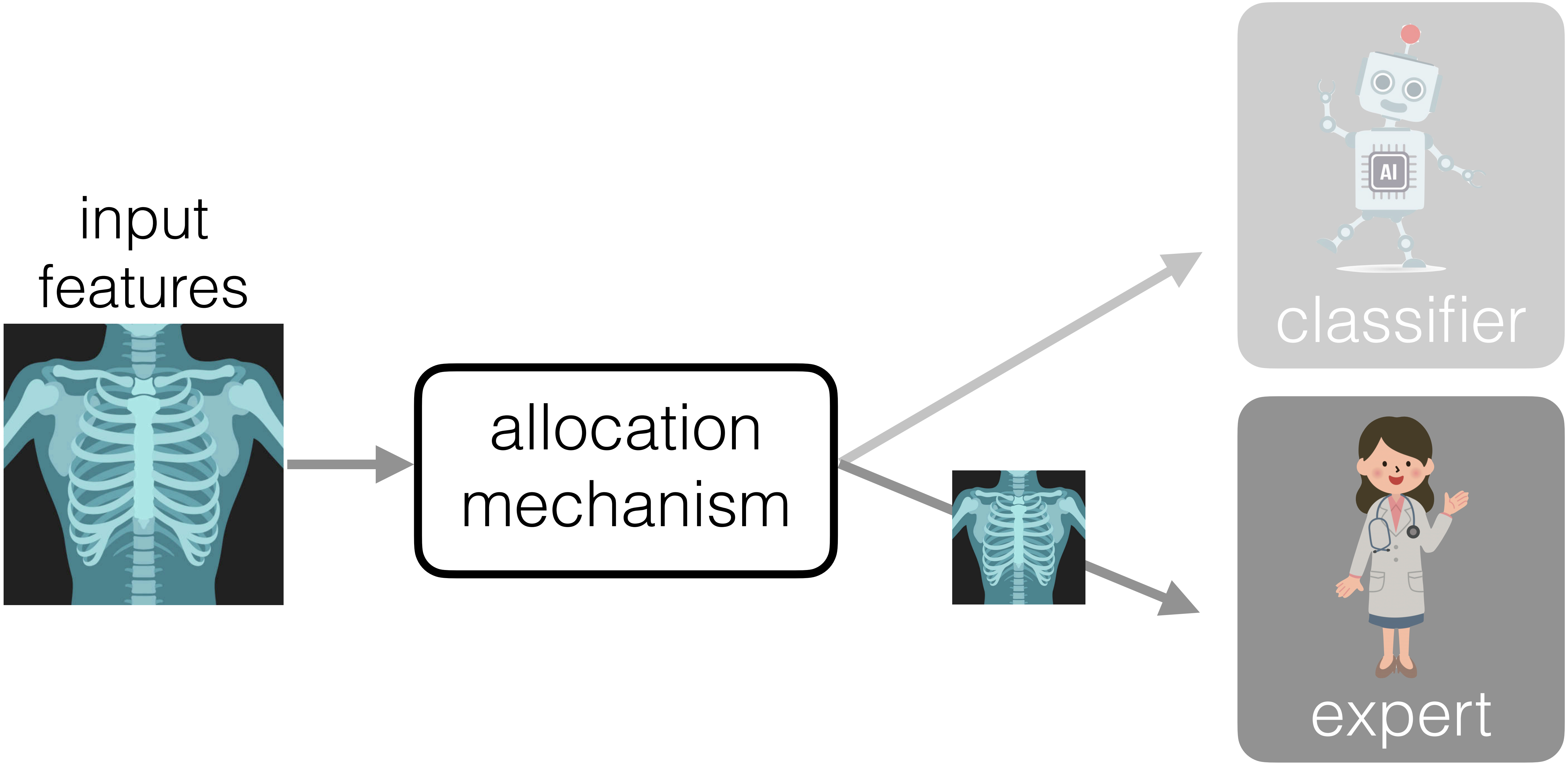
learning to defer (to an expert)



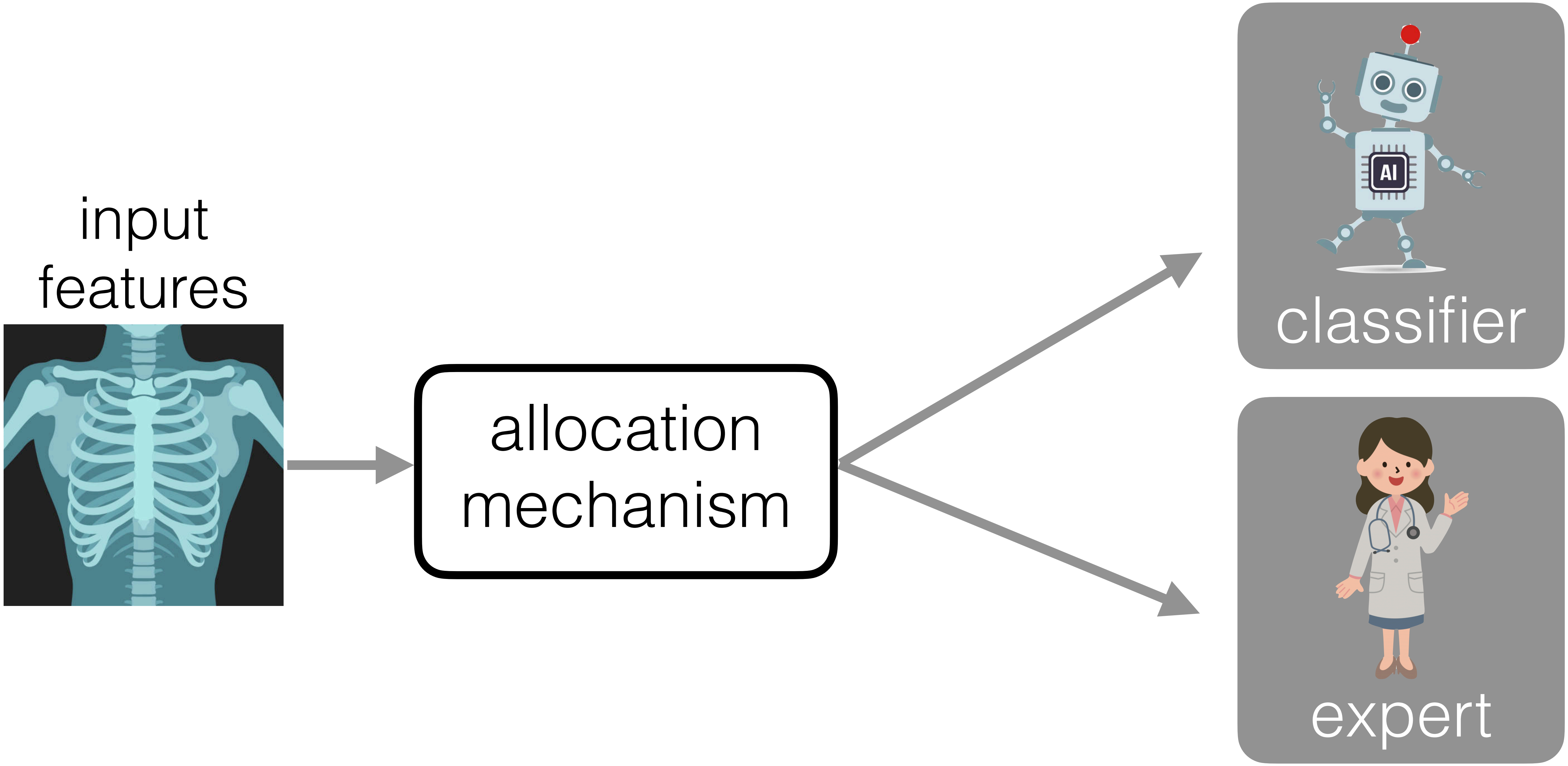
learning to defer (to an expert)



learning to defer (to an expert)

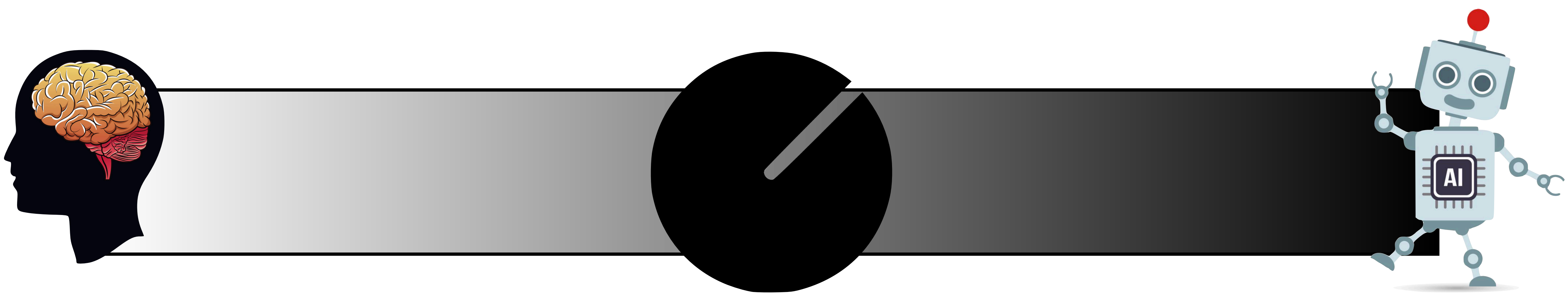


learning to defer (to an expert)



safe and robust semi-automation
via expert handling the hardest cases

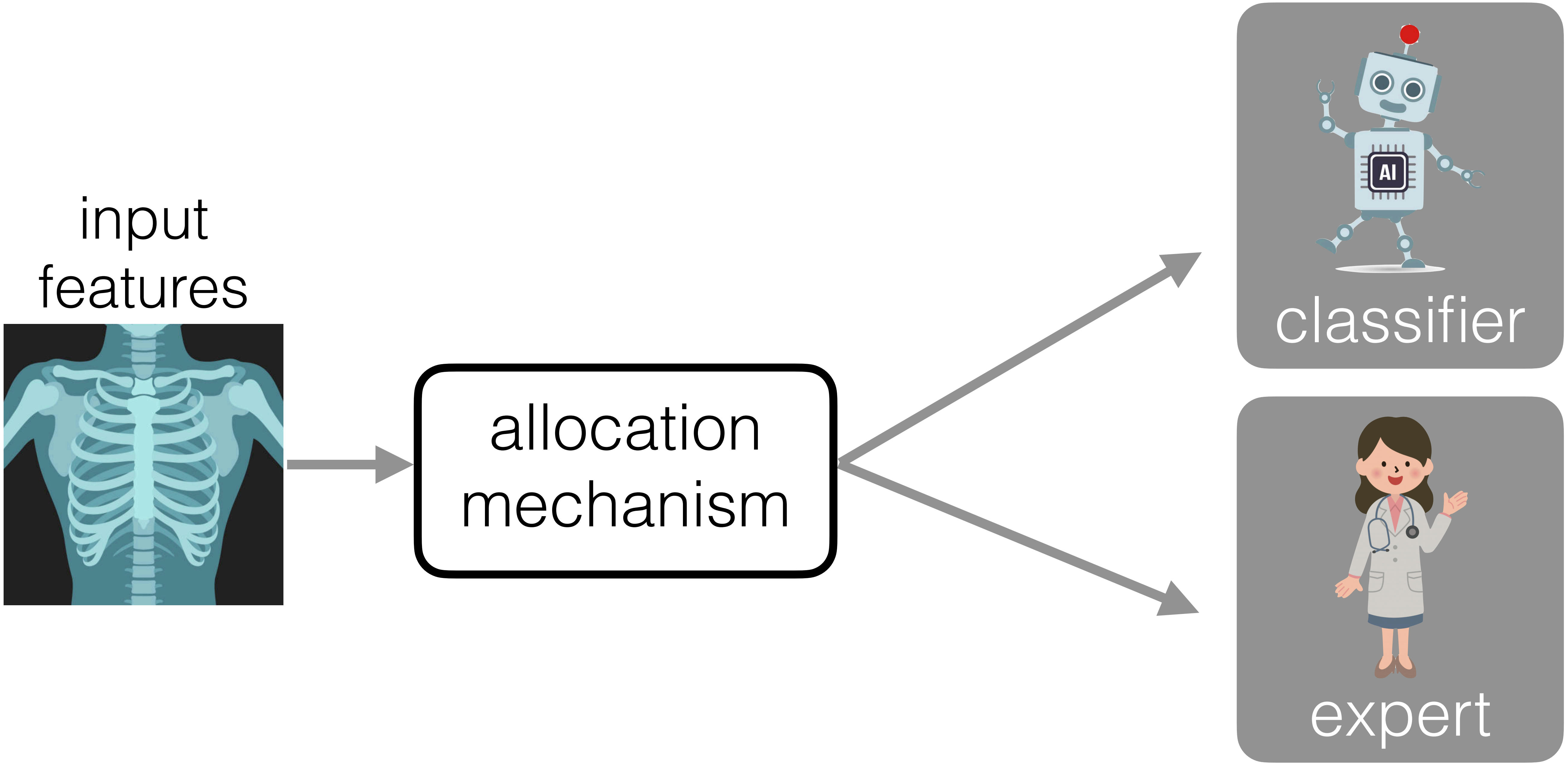
safe, gradual automation

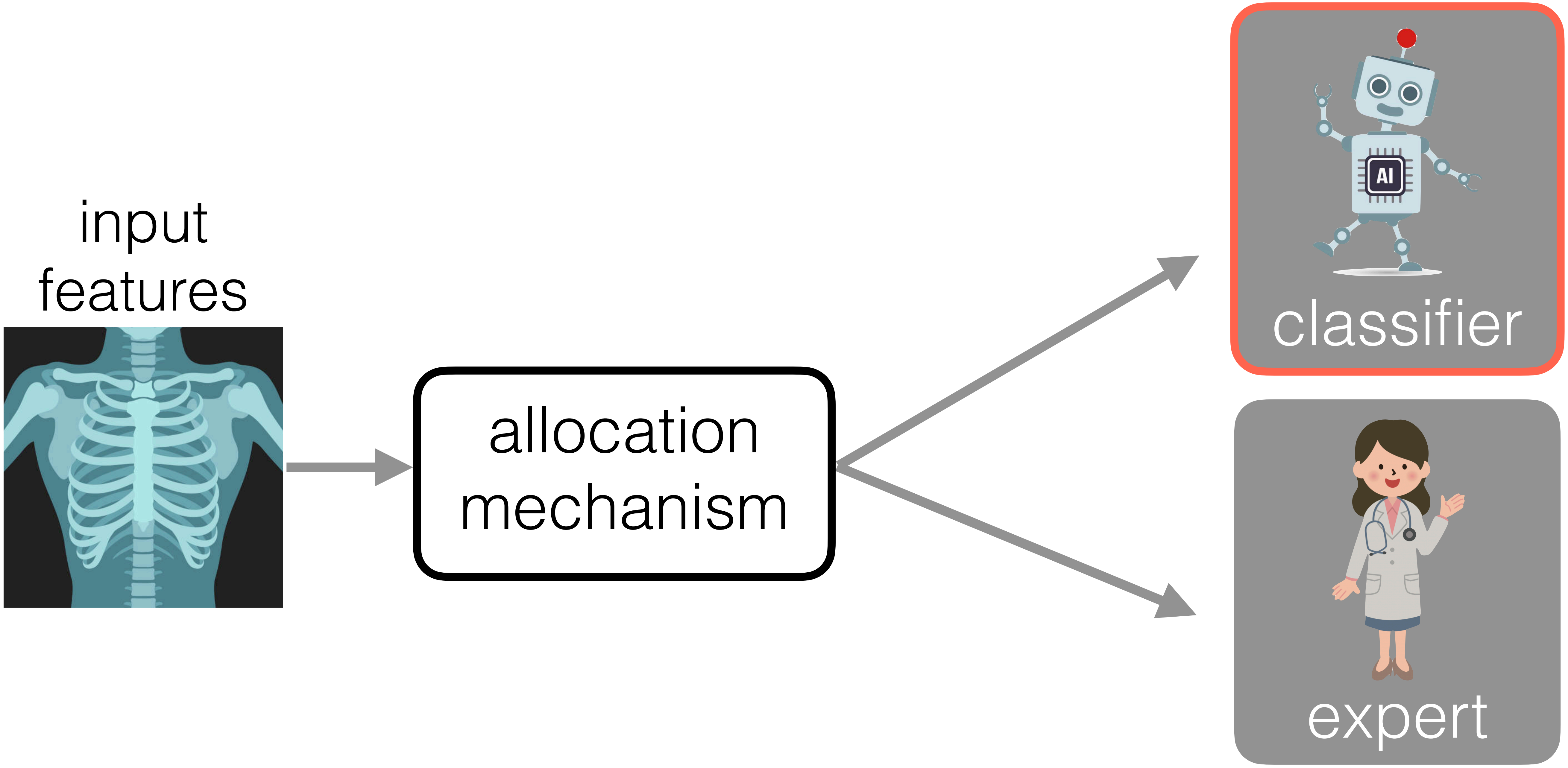


- single expert
  - softmax surrogate loss
  - improving calibration via one-vs-all
- multiple experts
  - surrogate losses
  - conformal sets of experts
- population of experts
  - surrogate losses
  - meta-learning a rejector

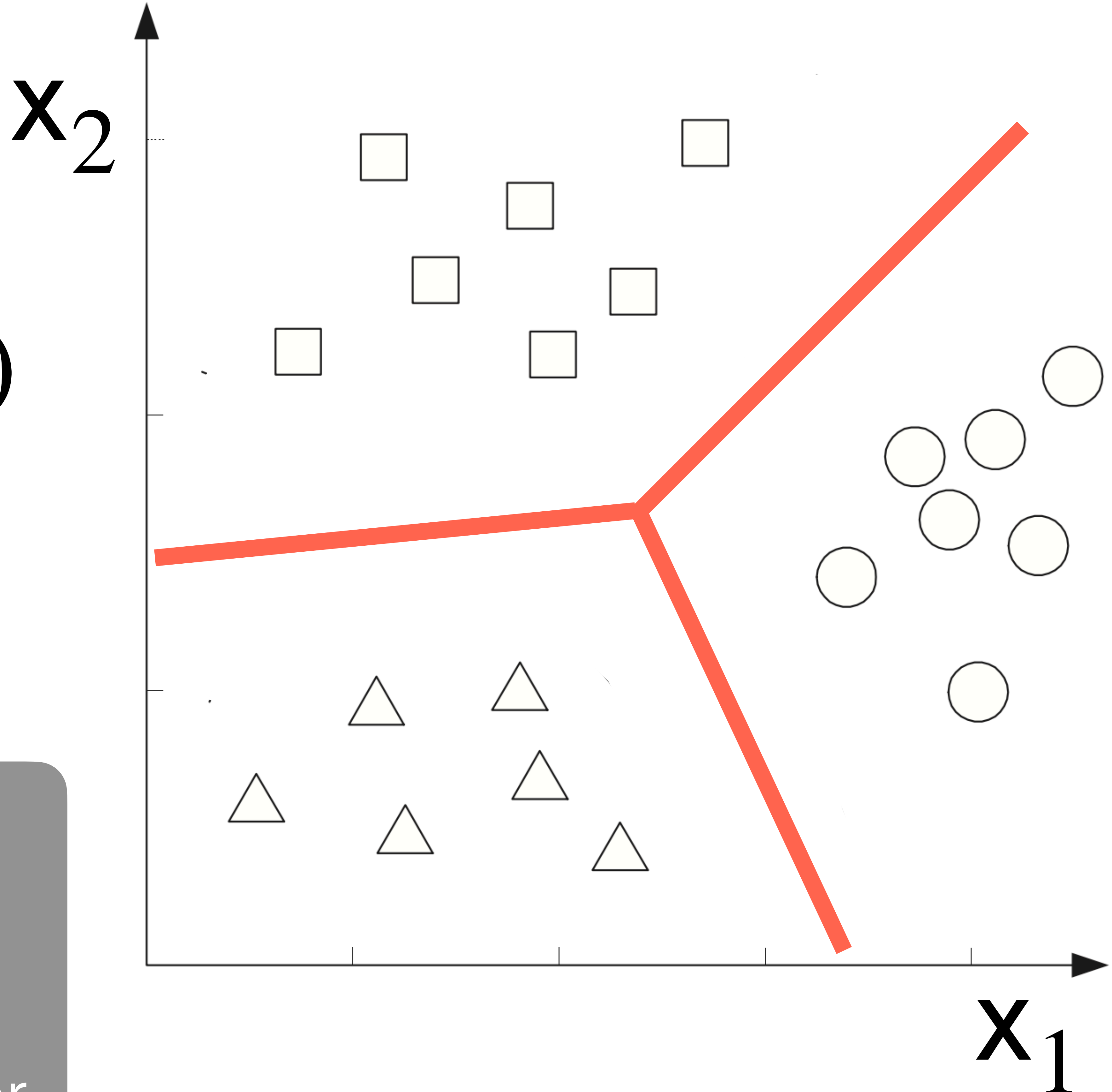
- ⊗ single expert
  - ⊗ softmax surrogate loss
  - ⊗ improving calibration via one-vs-all
- ⊗ multiple experts
  - ⊗ surrogate losses
  - ⊗ conformal sets of experts
- ⊗ population of experts
  - ⊗ surrogate losses
  - ⊗ meta-learning a rejector

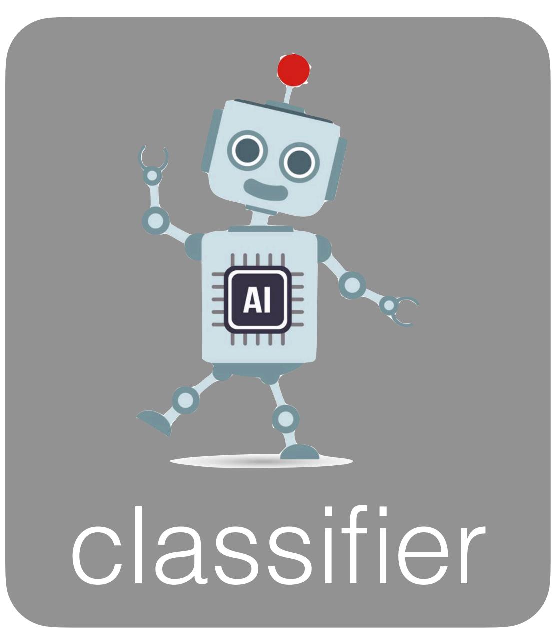
- ⊗ single expert
  - ⊗ softmax surrogate loss
  - ⊗ improving calibration via one-vs-all
- ⊗ multiple experts
  - ⊗ surrogate losses
  - ⊗ conformal sets of experts
- ⊗ population of experts
  - ⊗ surrogate losses
  - ⊗ meta-learning a rejector


input
features
allocation
mechanism
AI
classifier
expert

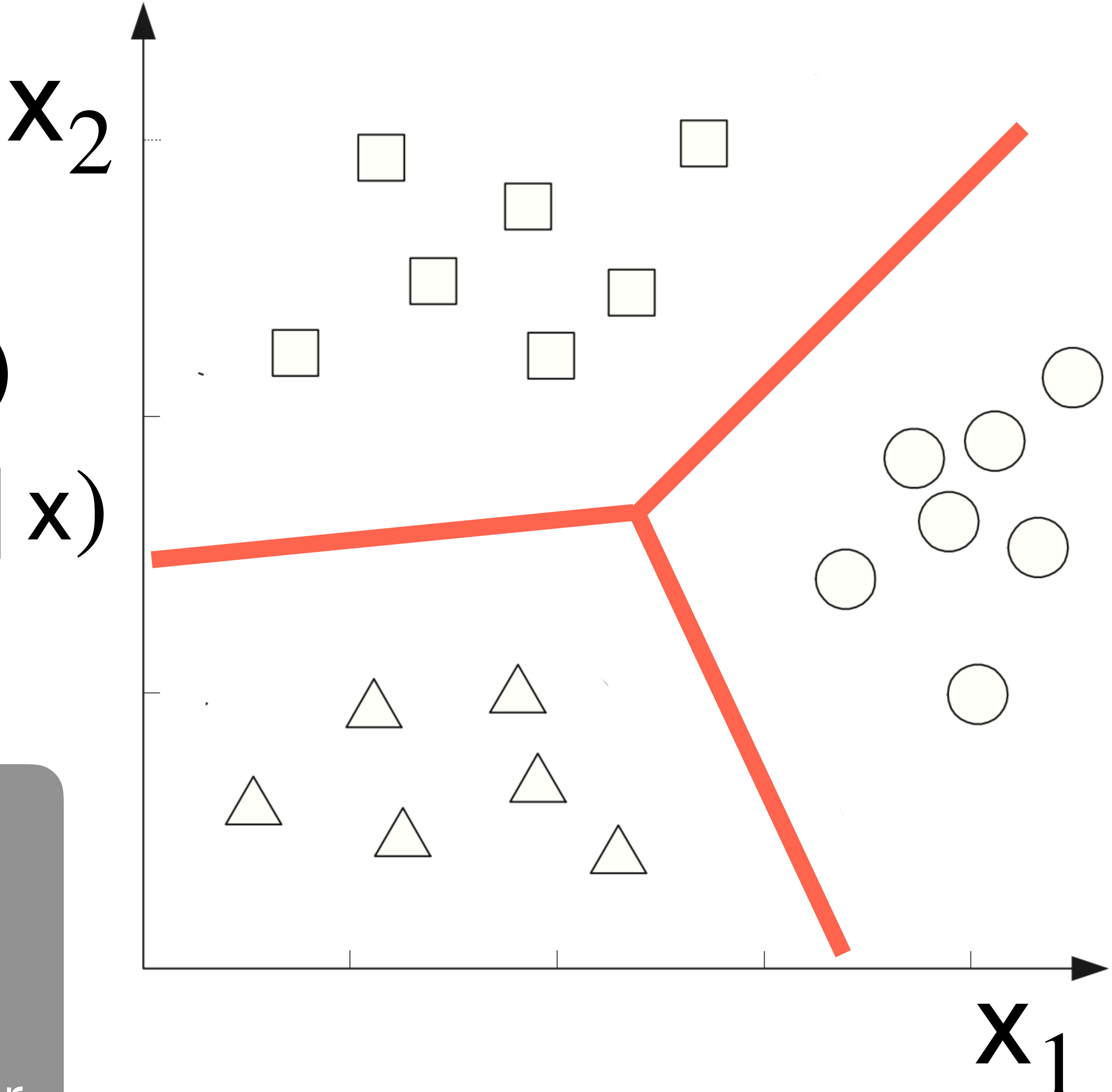

input
features
allocation
mechanism
classifier
AI
expert

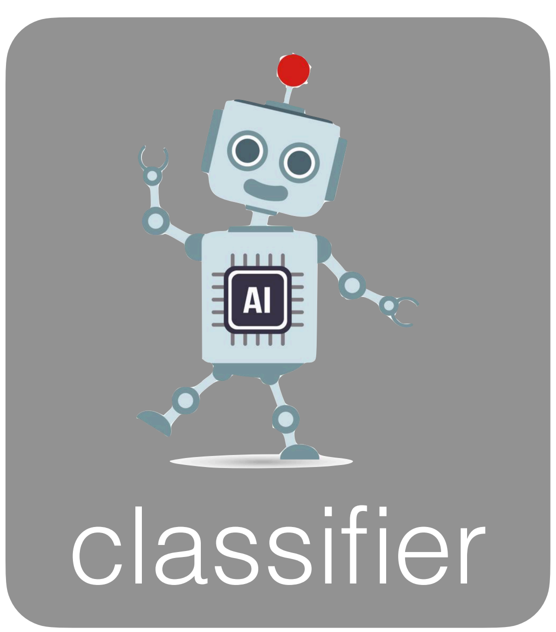
$p(y \mid x)$

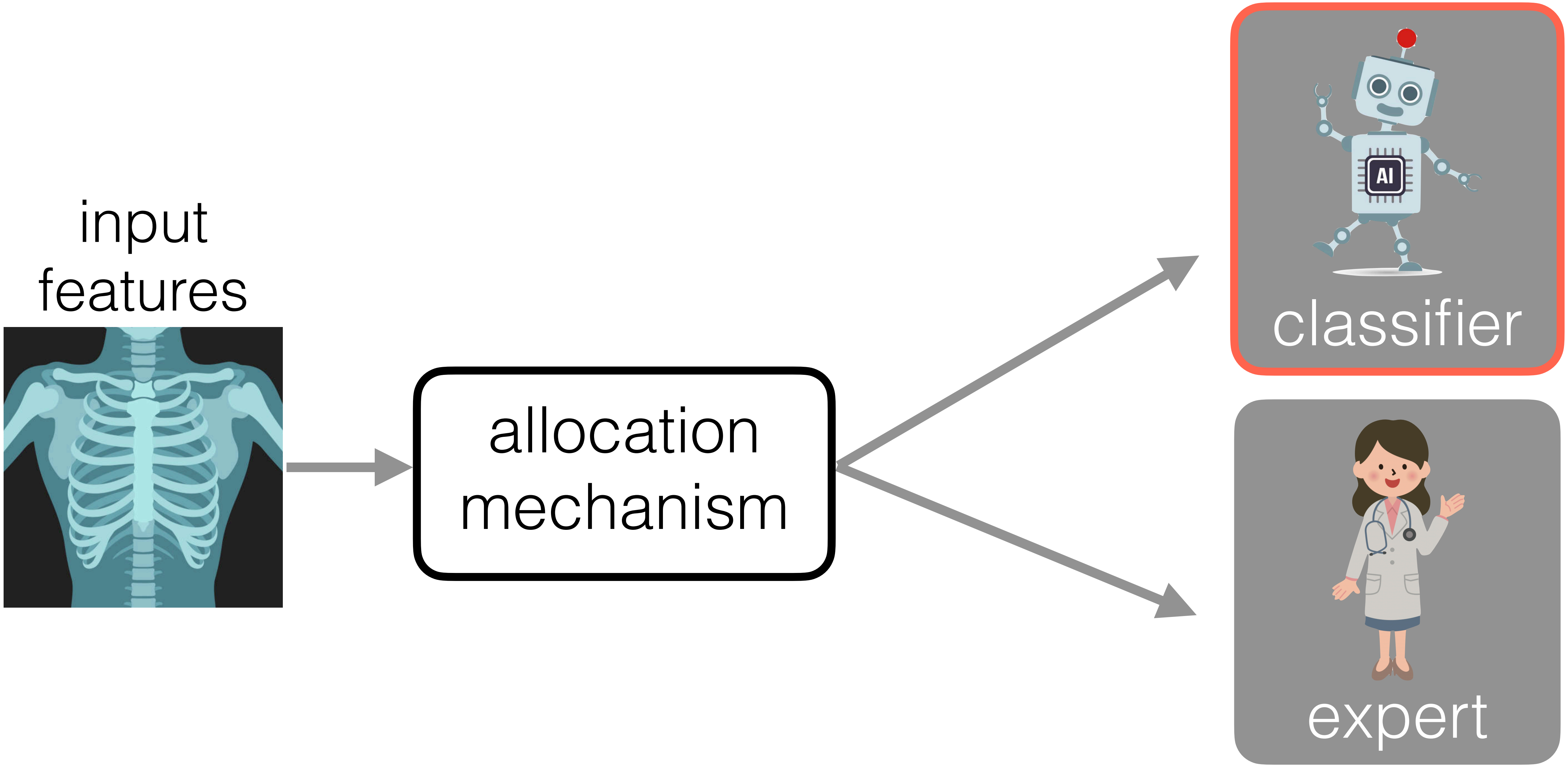


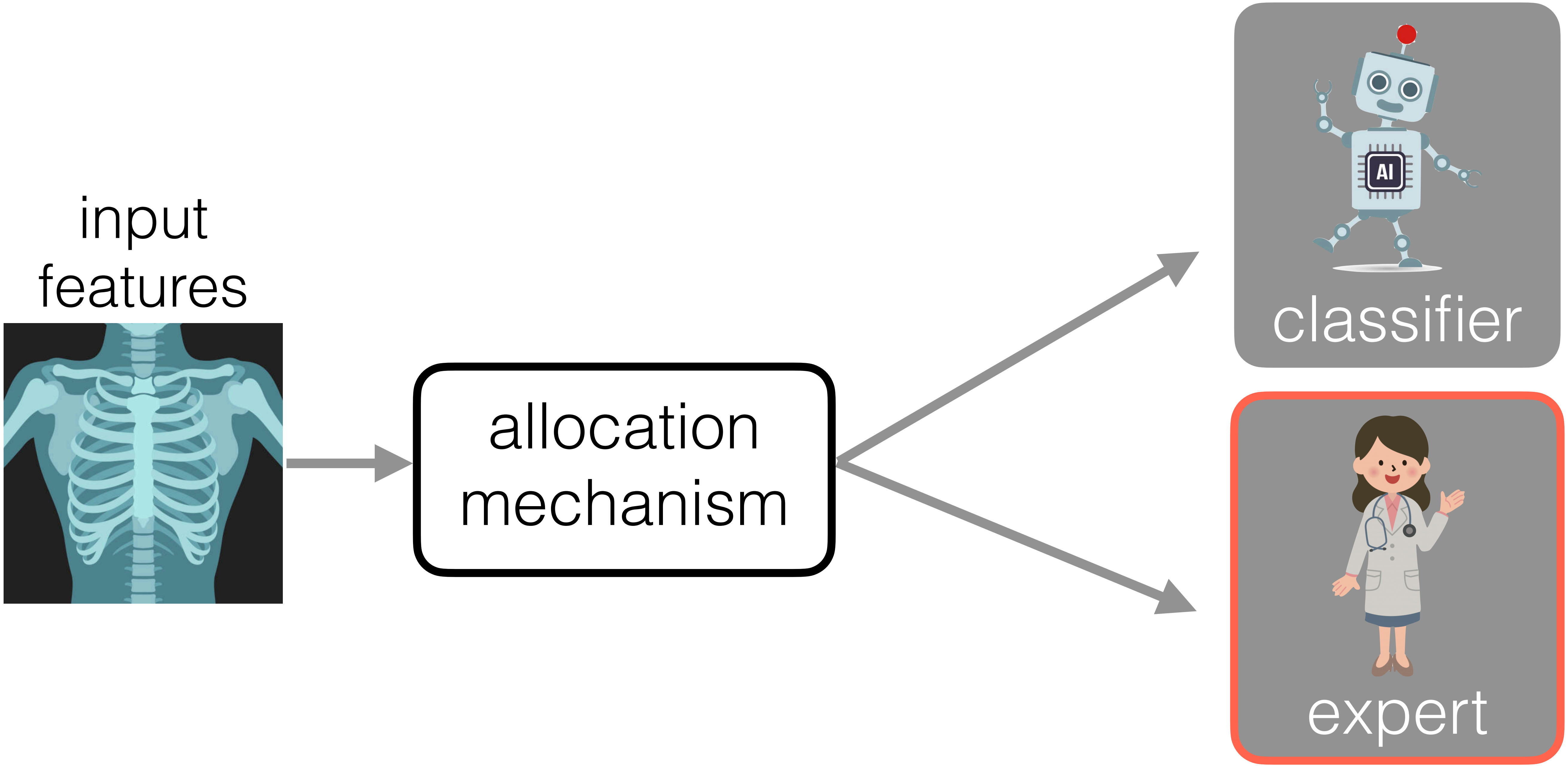


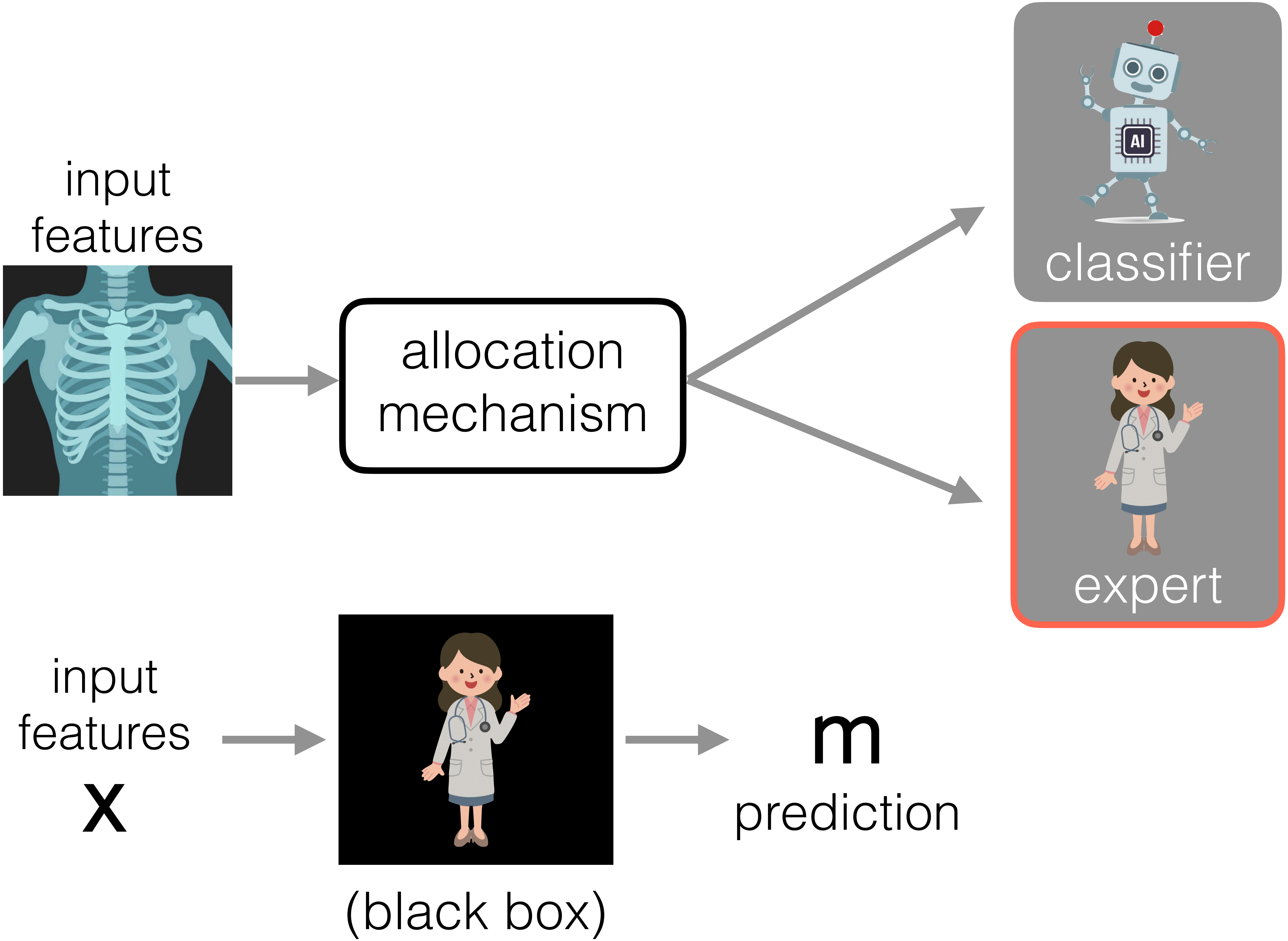
$$p(y\,|\,x) \approx \mathbb{P}(y\,|\,x)$$

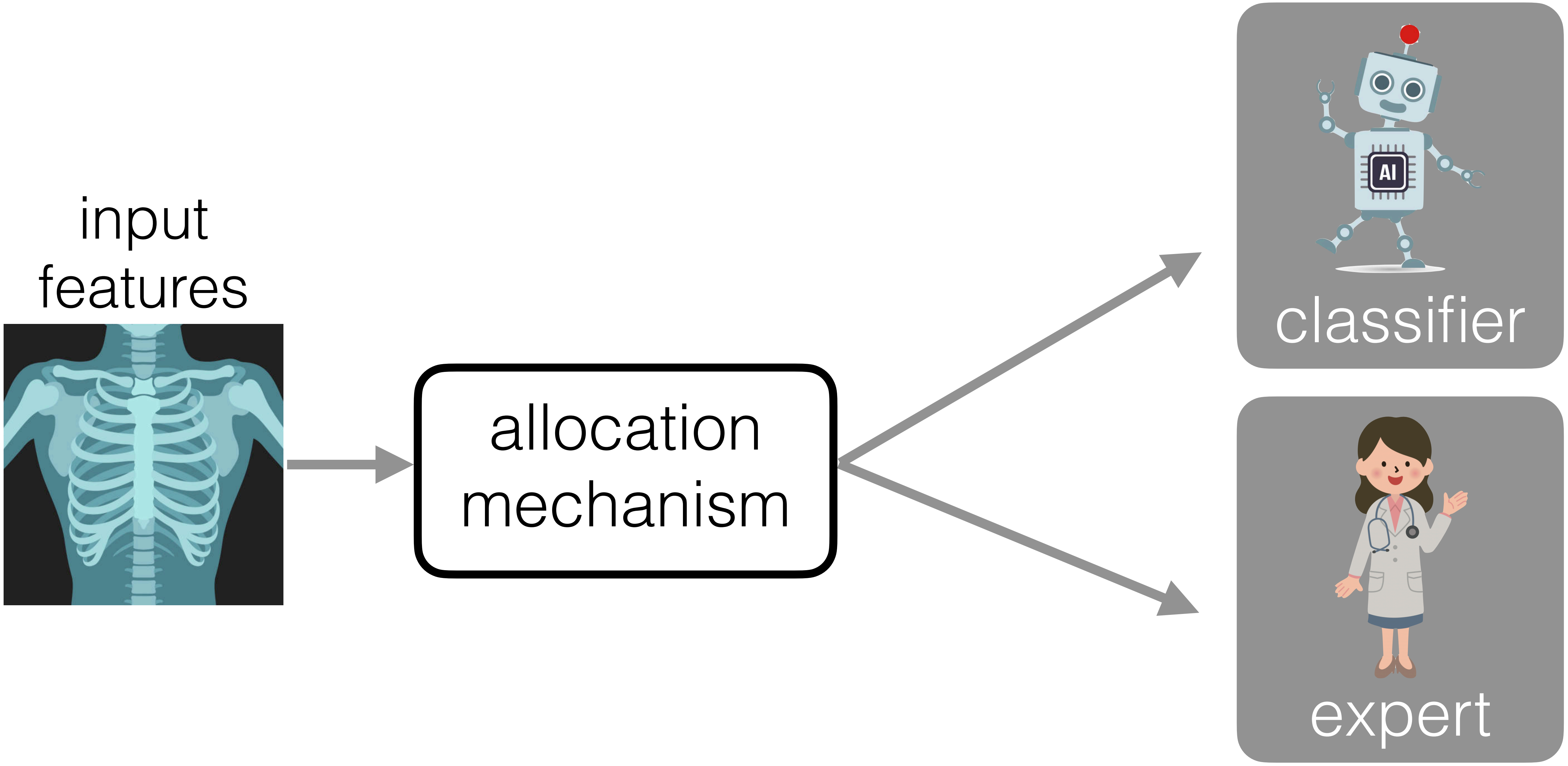


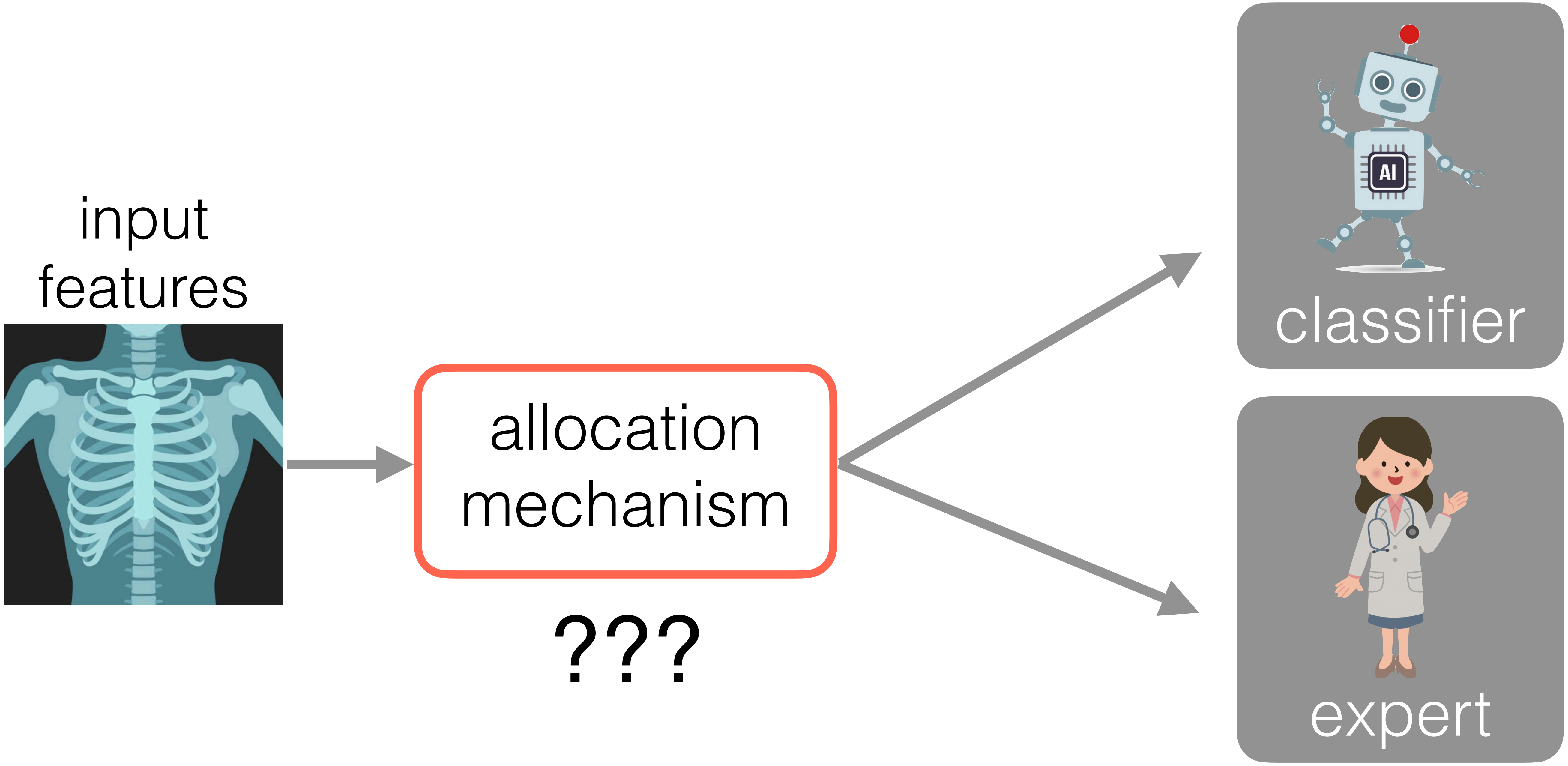



input
features
allocation
mechanism
AI
classifier
expert

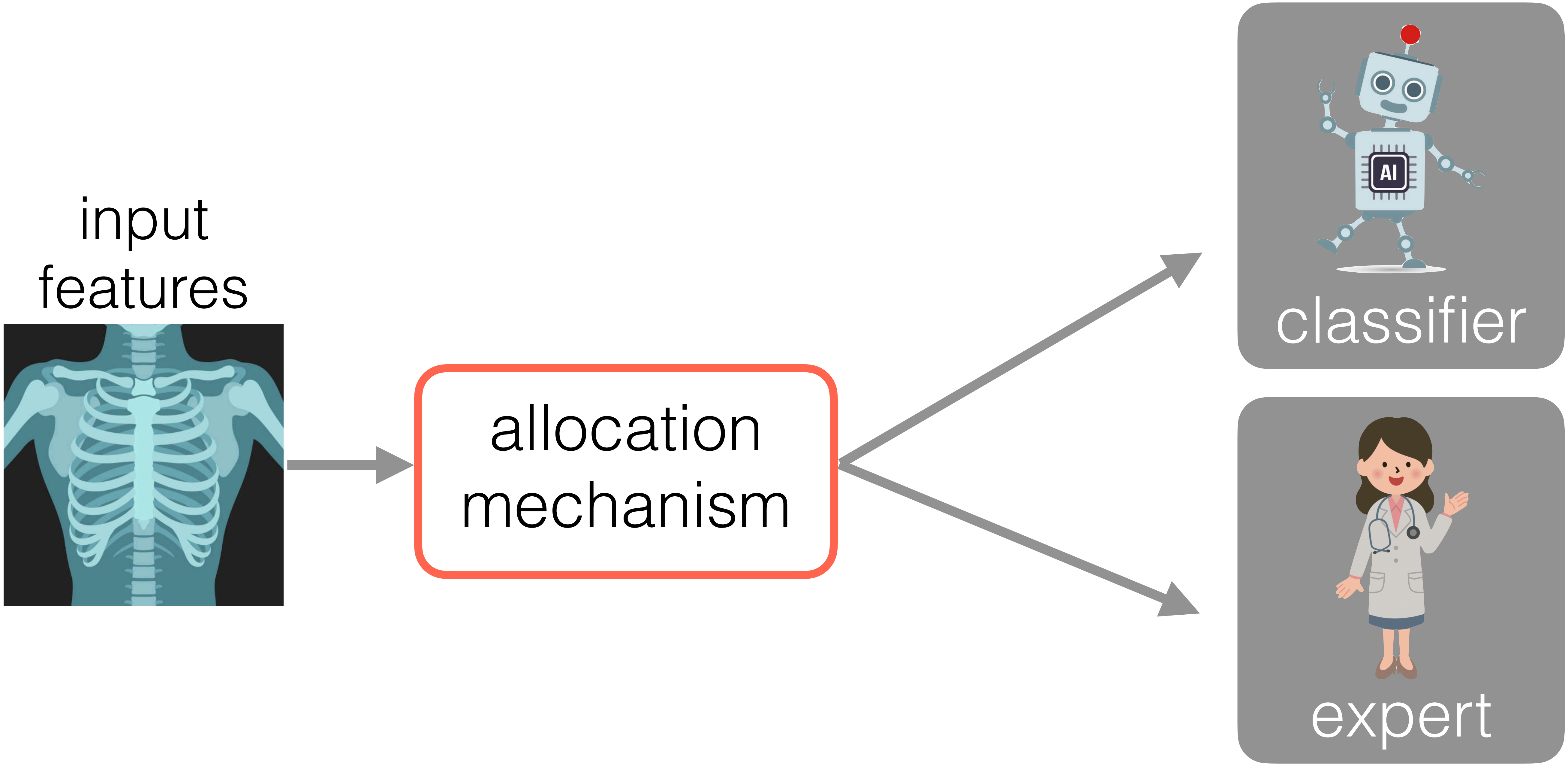

input
features
allocation
mechanism
AI
classifier
expert


input
features
allocation
mechanism
classifier
expert
input
features
x
(black box)
m
prediction


input
features
allocation
mechanism
classifier
AI
expert



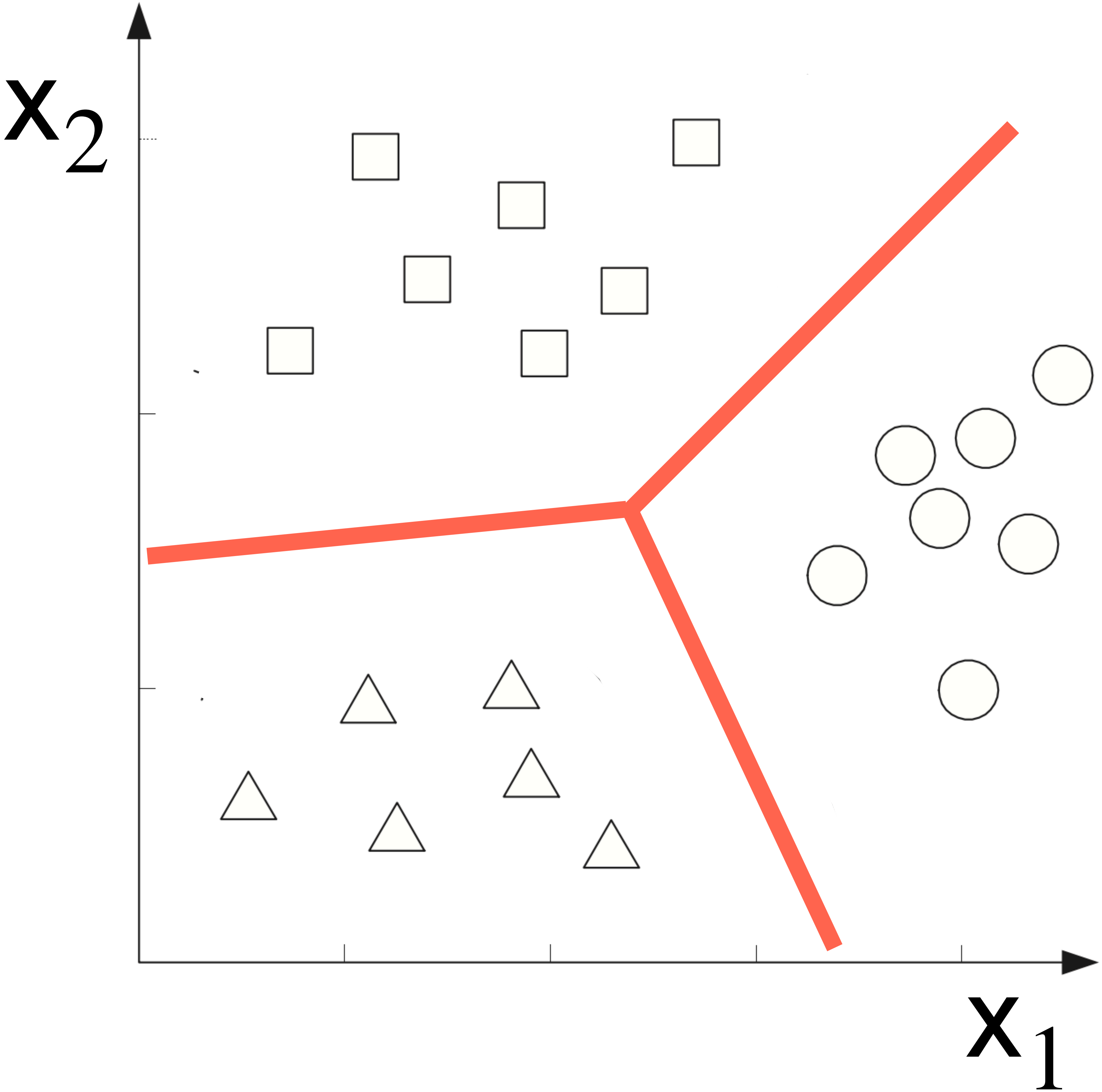
input
features
allocation
mechanism
???
AI
classifier
expert

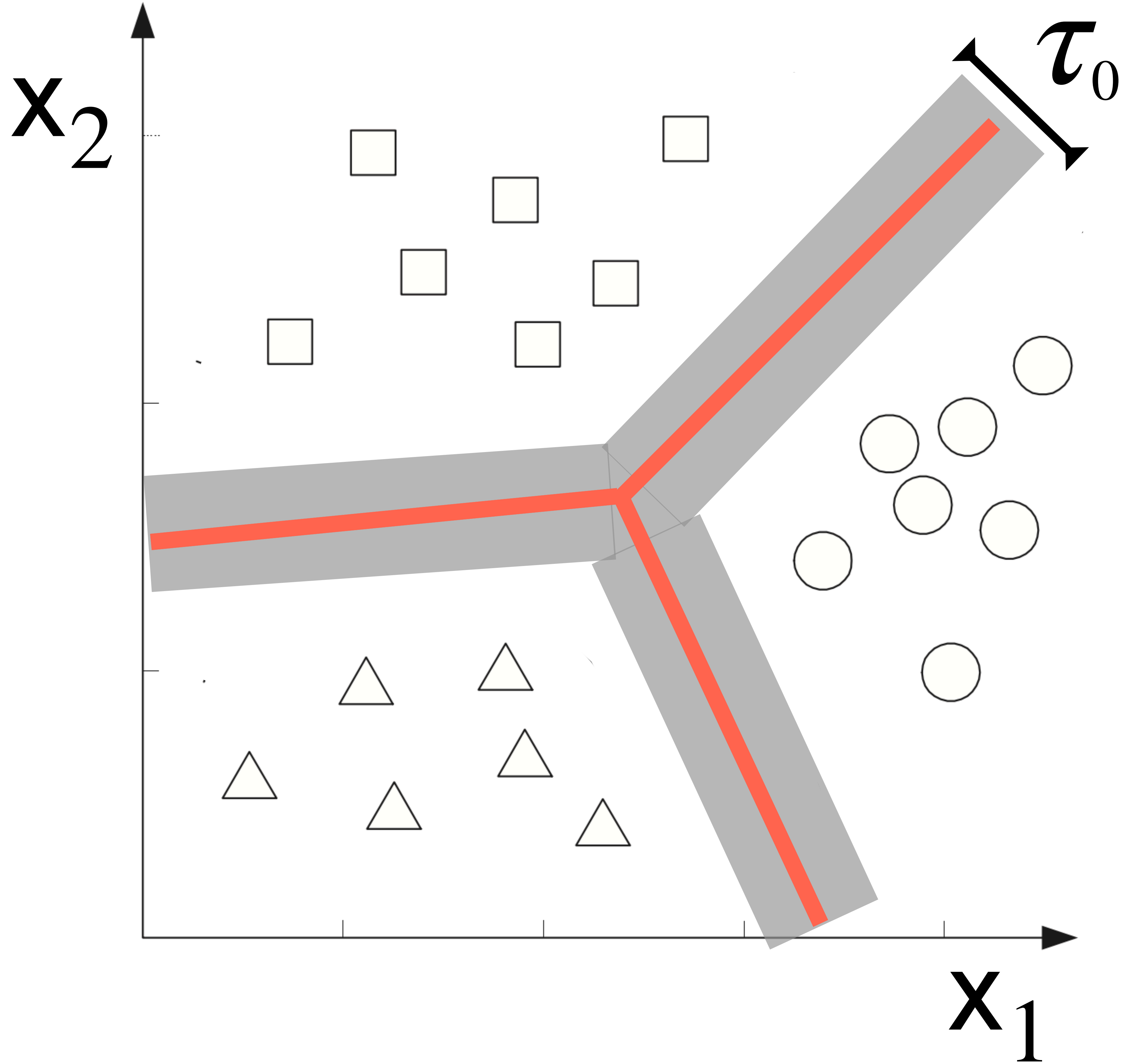


defer to expert if…

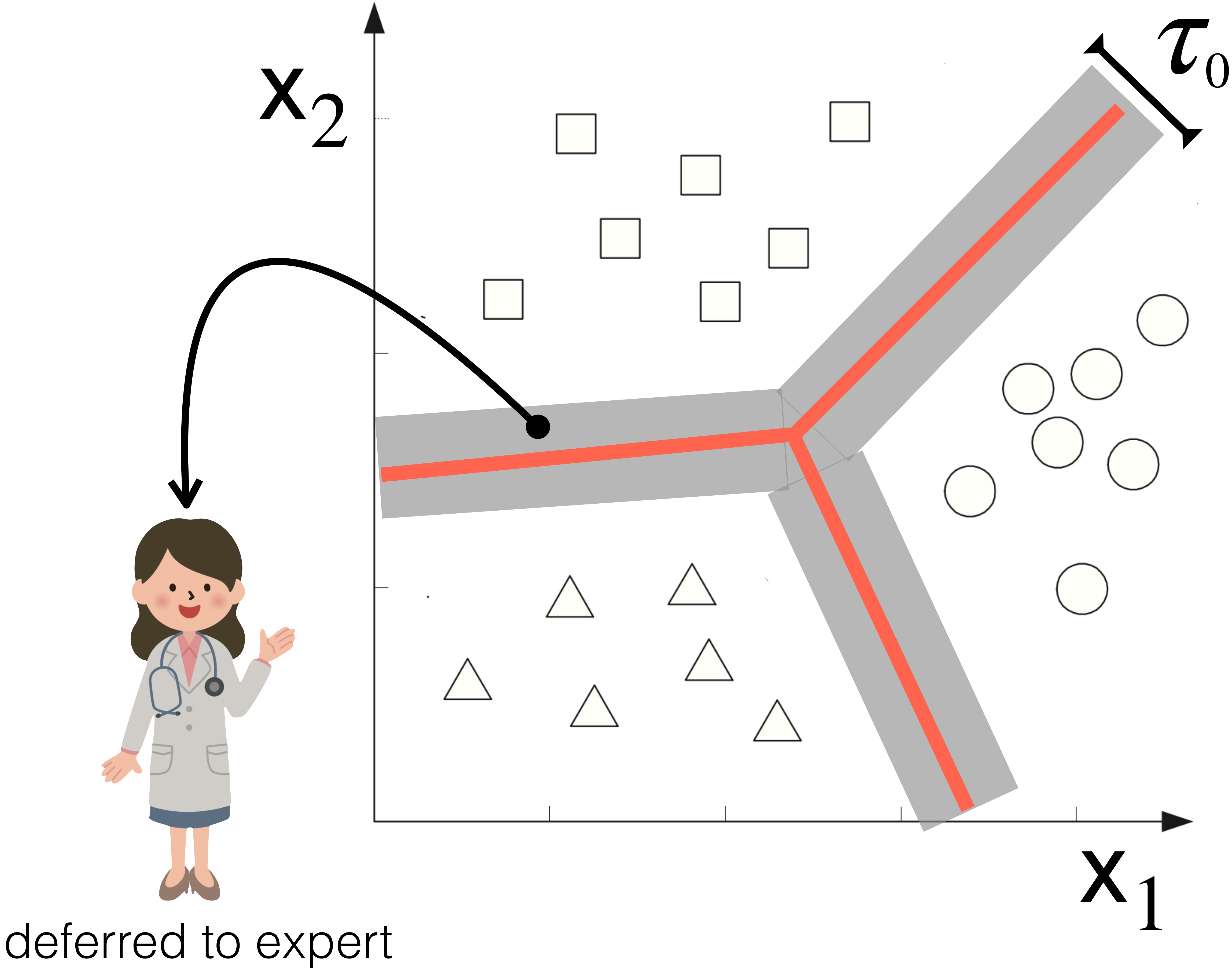
$$\max_{y} \; \mathrm{p}(y \mid \mathrm{x}) \;\leq\; \tau_0$$

(constant)

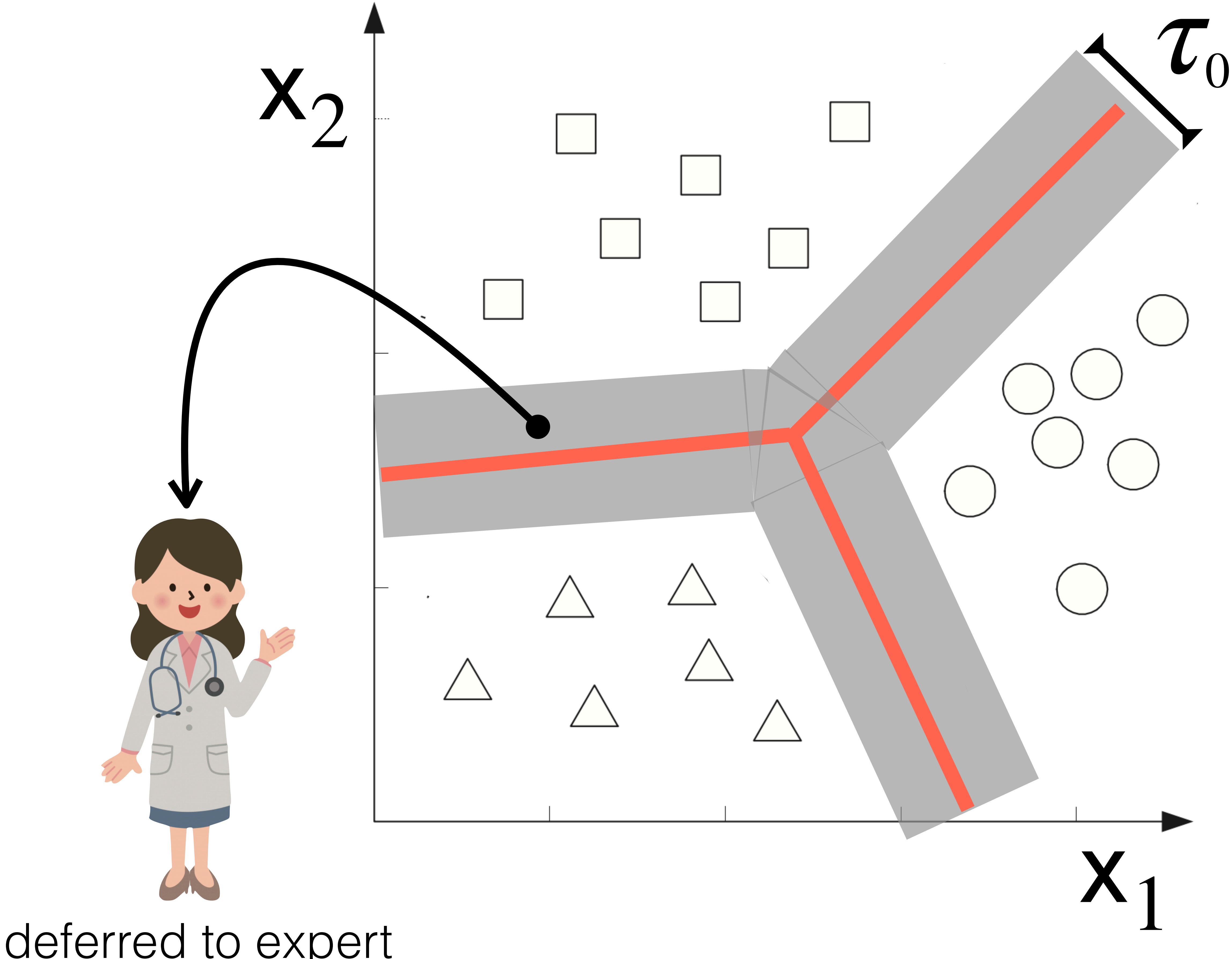

x_2
x_1



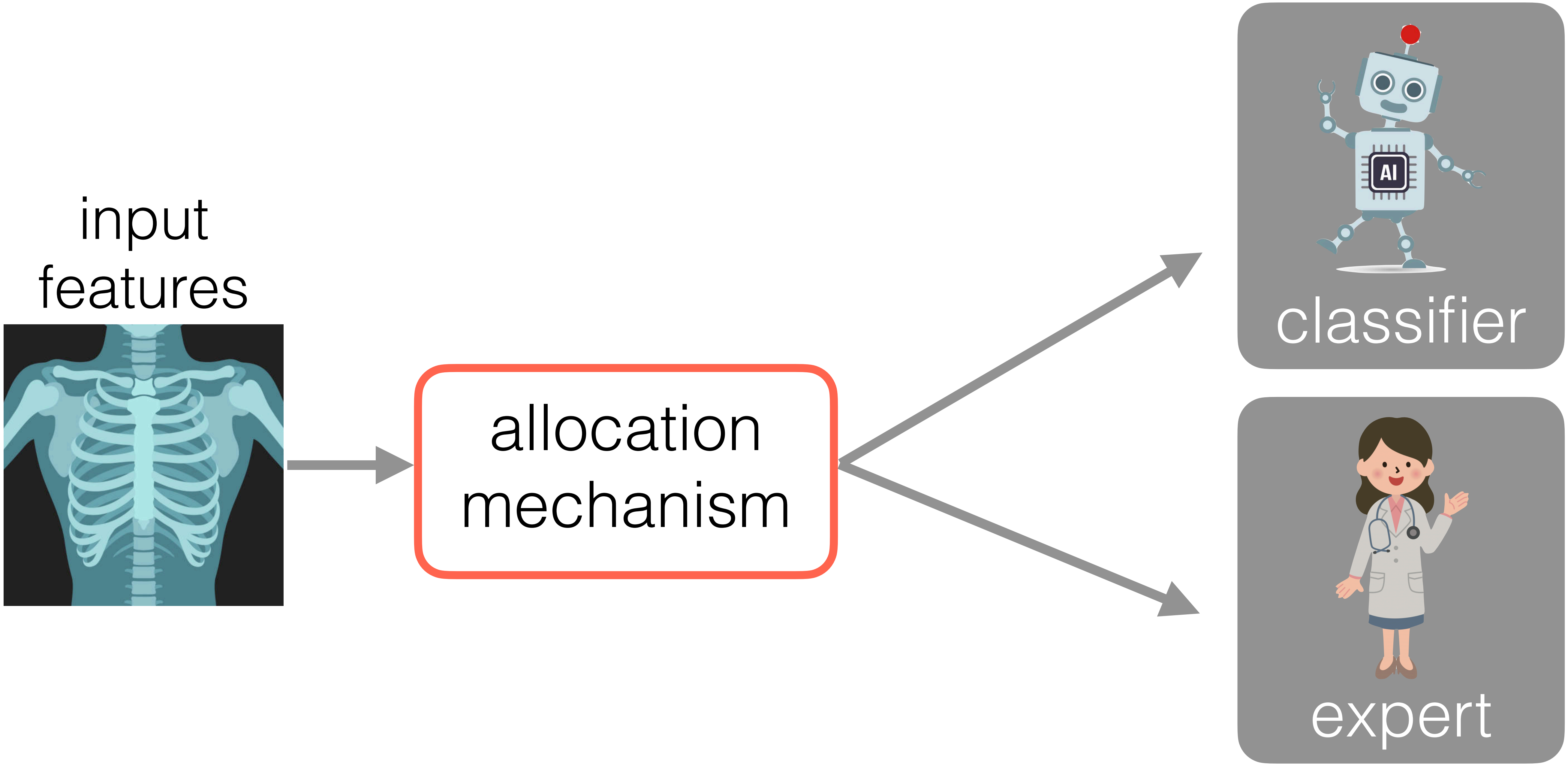
τ_0
x_2
x_1



τ_0
x_2
x_1
deferred to expert



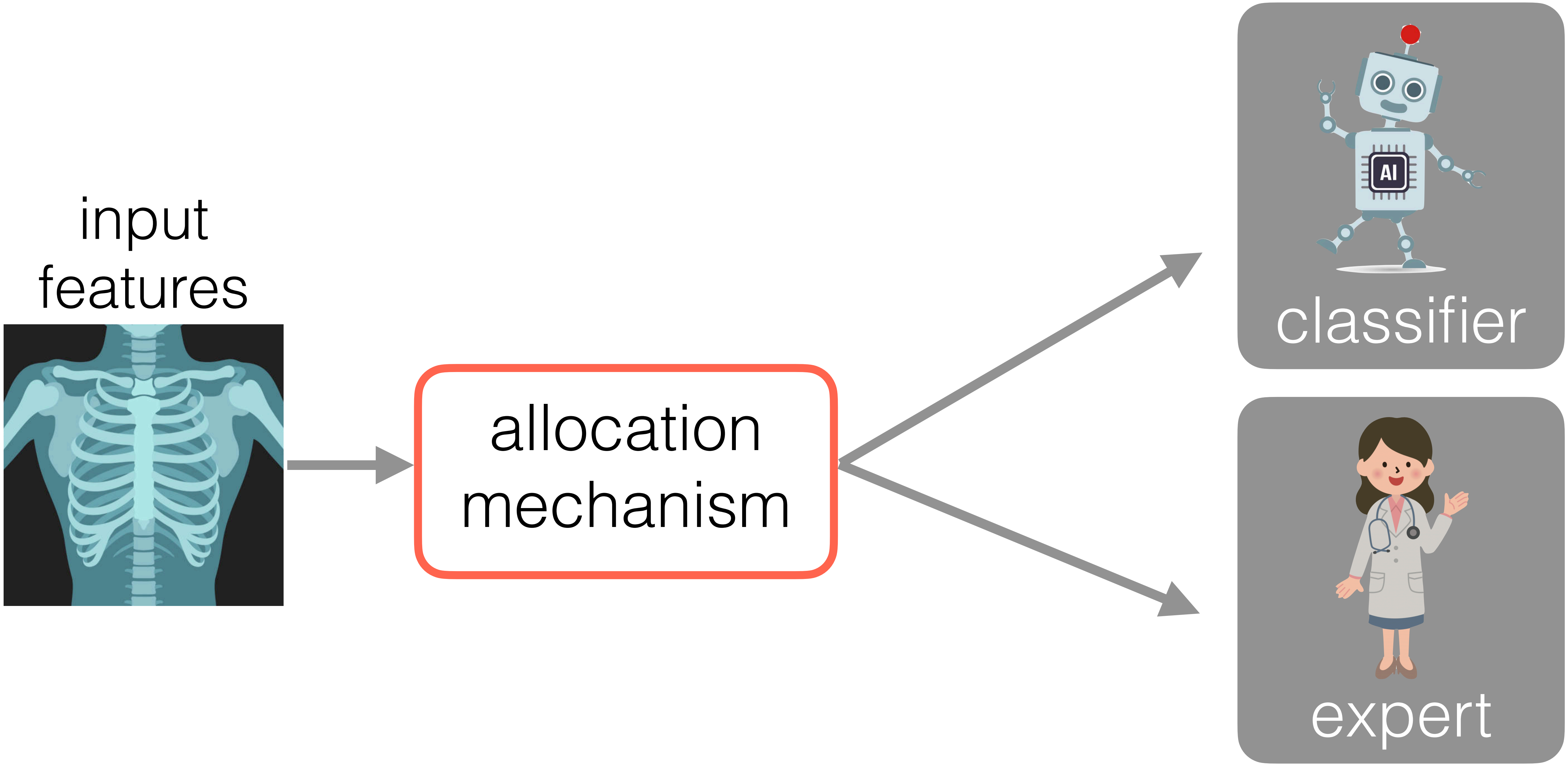
x_2
τ_0
x_1
deferred to expert



defer to expert if…

$$\max_{y} \; \mathrm{p}(y \mid \mathrm{x}) \;\leq\; \tau_0$$

(constant)

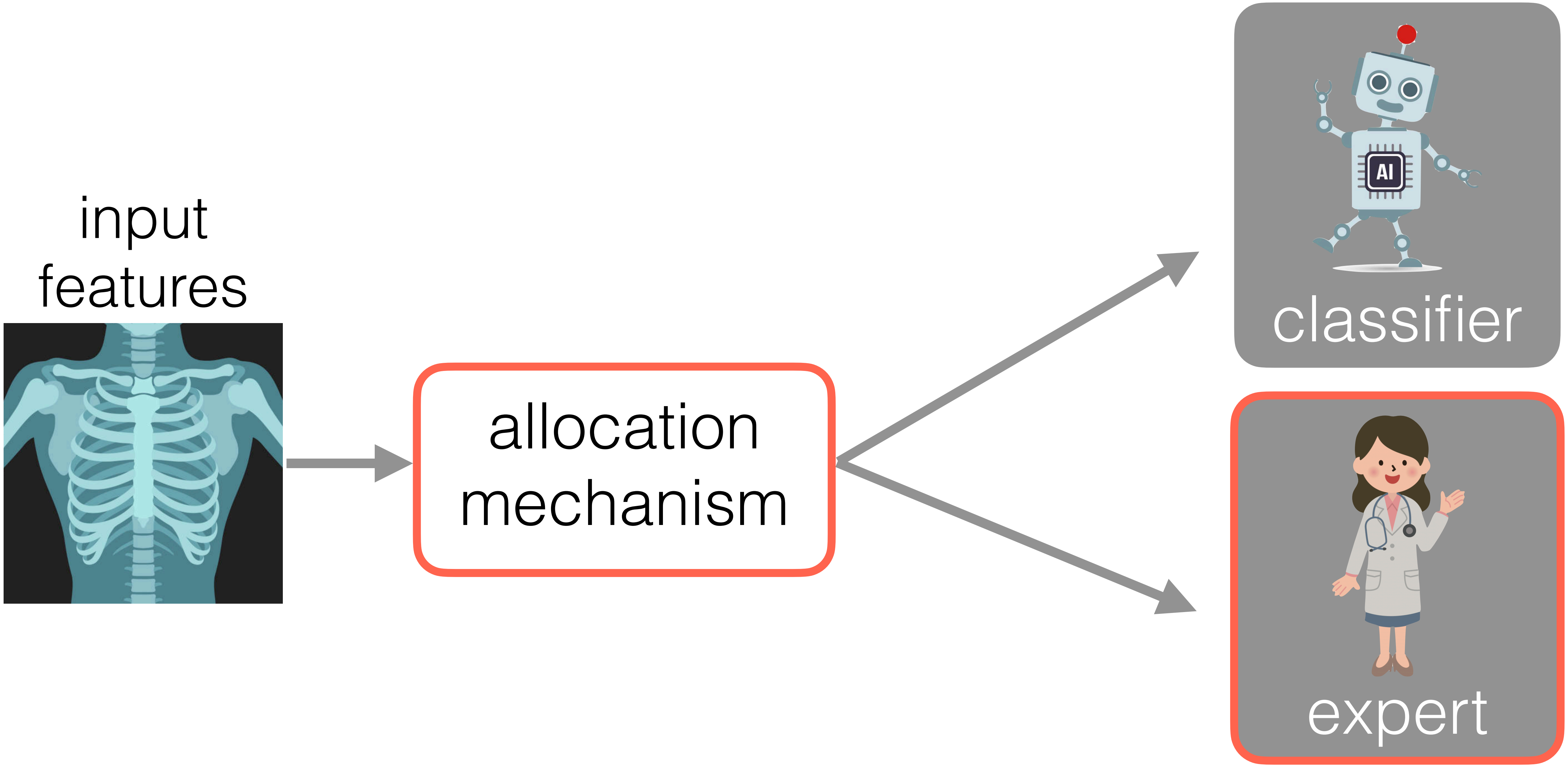


defer to expert if…

$$\max_{y} \; \mathrm{p}(y \,|\, \mathrm{x}) \;\leq\; \tau_0$$

(constant)

problem?

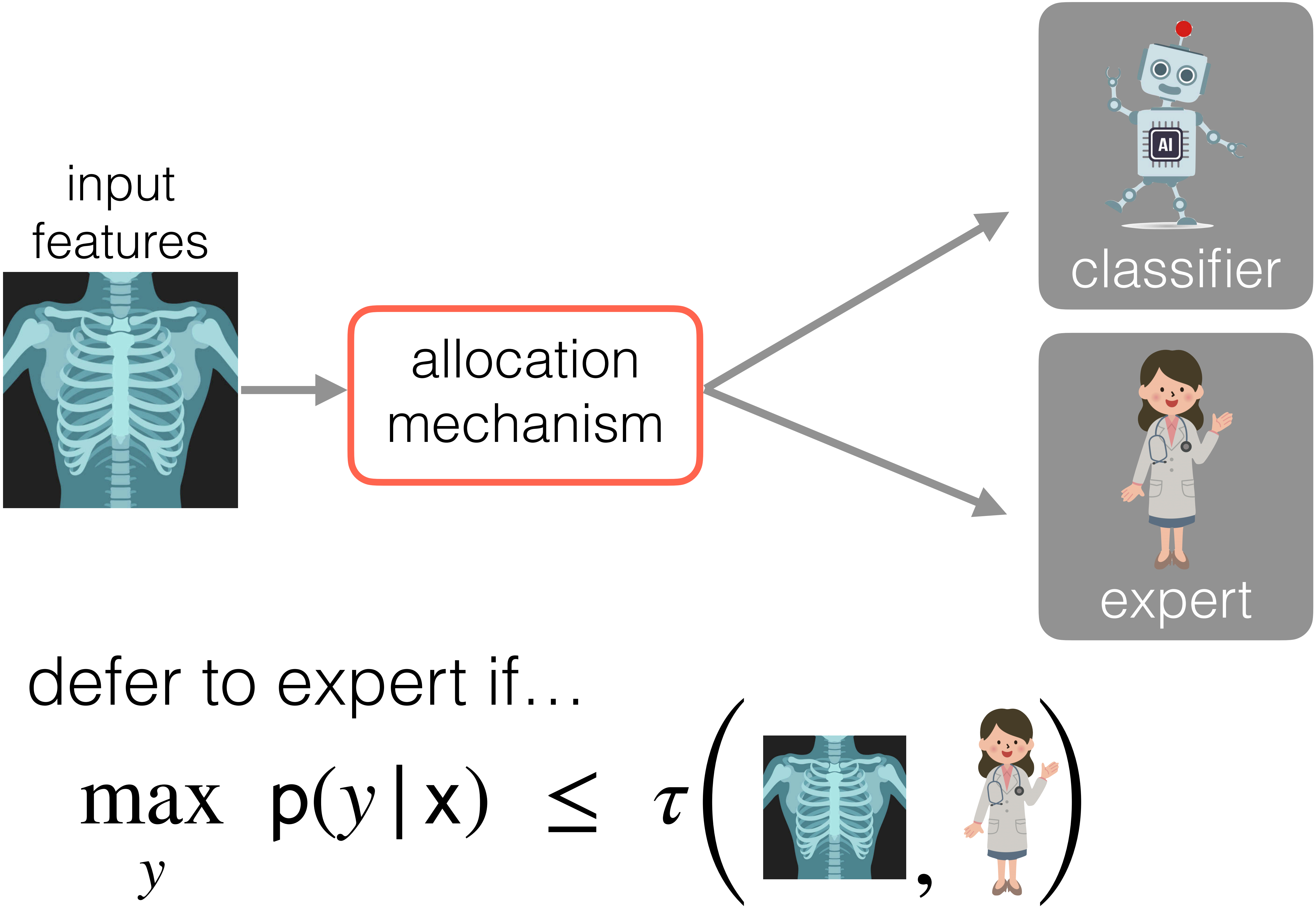


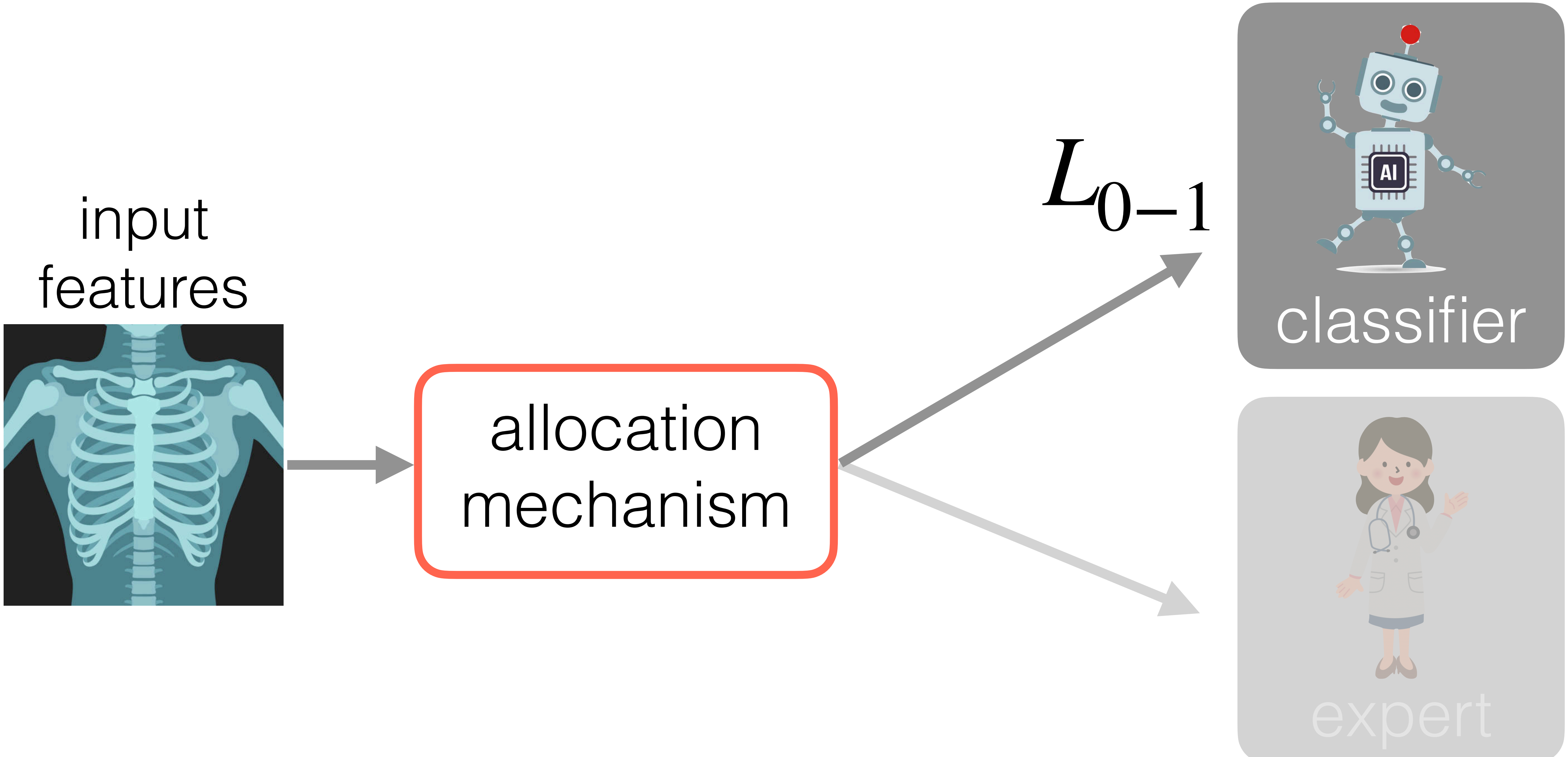
defer to expert if…

$$\max_{y} \; \mathrm{p}(y \,|\, \mathrm{x}) \;\leq\; \tau_0$$

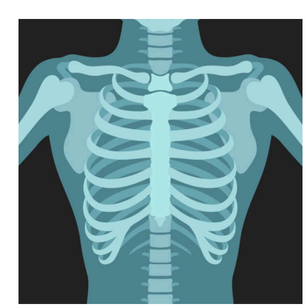
(constant)

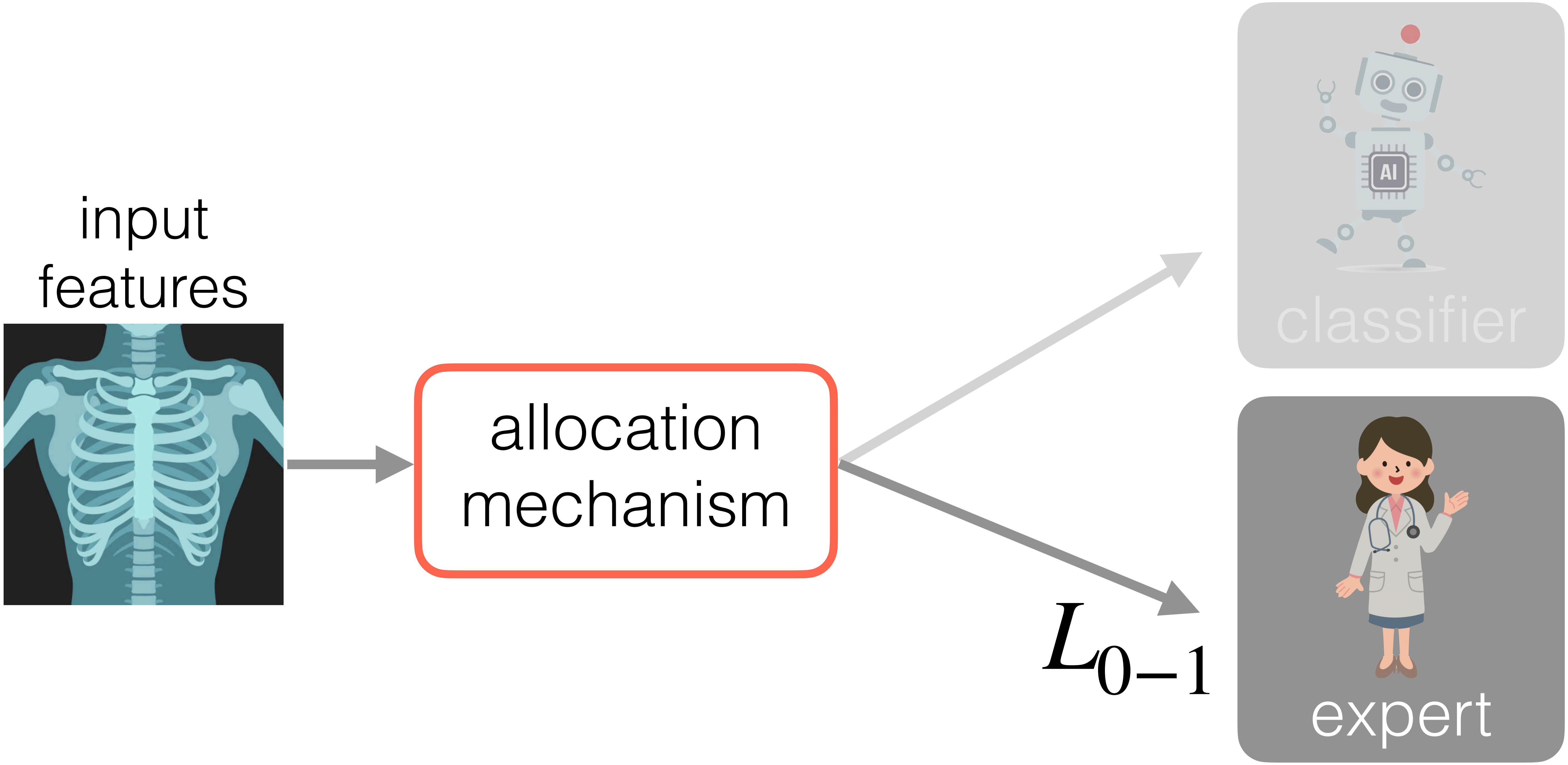
the expert's knowledge is not considered!


input features
allocation mechanism
classifier
AI
expert
defer to expert if…
max p(y | x) ≤ τ(,)
y

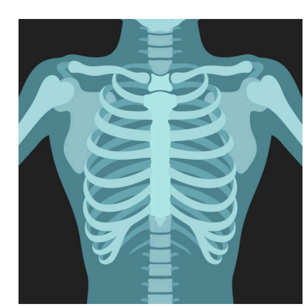


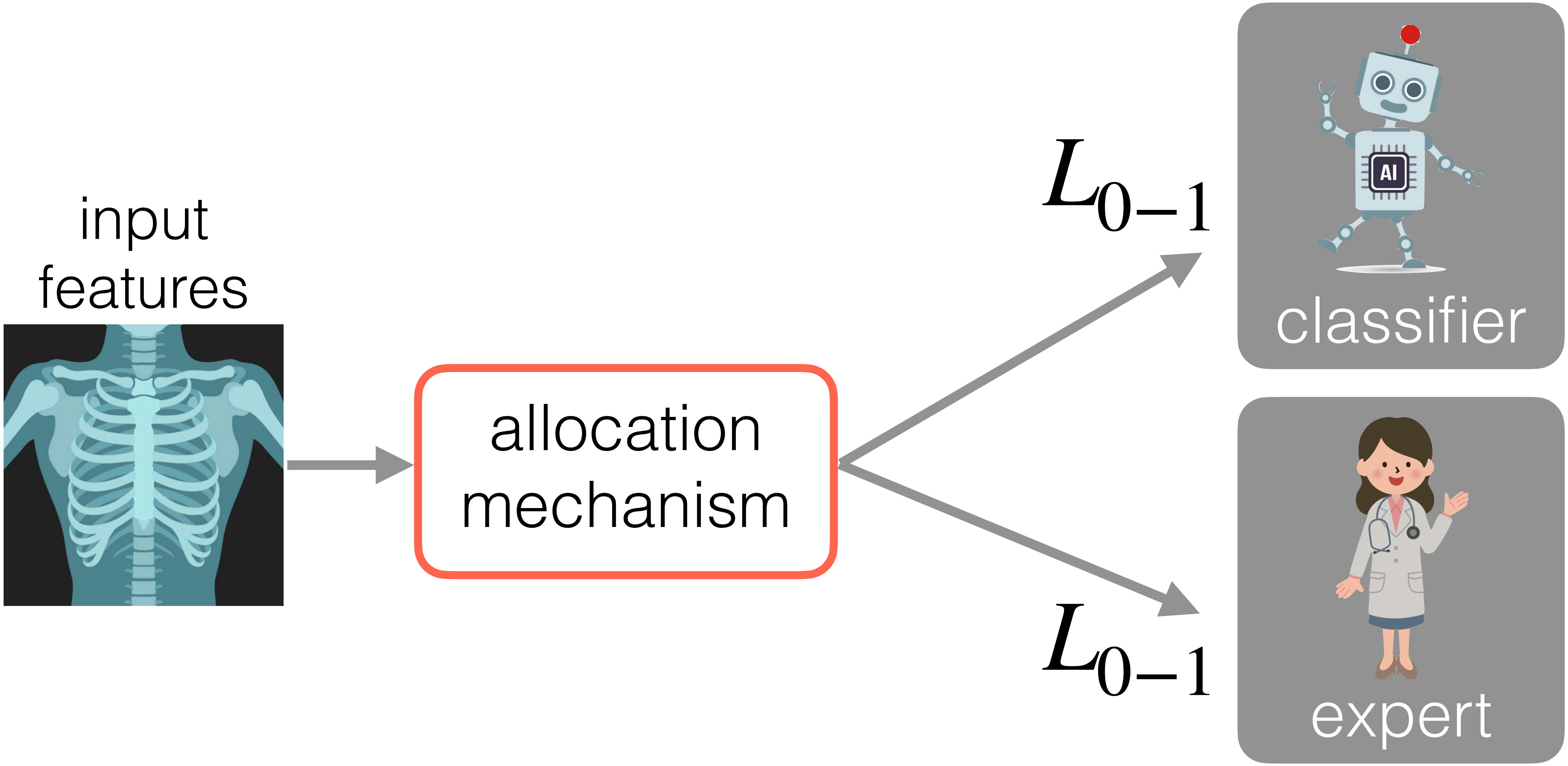
defer to expert if…

$\max_{y} \; \mathrm{p}(y \mid \mathrm{x}) \;\leq\; \tau\Big($ ,  $\Big)$



defer to expert if…

$\max_{y}\ \mathrm{p}(y \mid \mathrm{x}) \ \leq\ \tau\Big($ ,  $\Big)$



Bayes optimal deferral rule:

$$\max_{y} \mathbb{P}(y \mid \mathrm{x}) \leq \mathbb{P}(\mathrm{m} = \mathrm{y} \mid \mathrm{x})$$

probability that the expert is correct

[Mozannar & Sontag, 2020]

softmax implementation [Mozannar & Sontag, 2020]

softmax implementation

[Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n\}_{n=1}^{N}$$

softmax implementation

[Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n\}_{n=1}^{N}$$

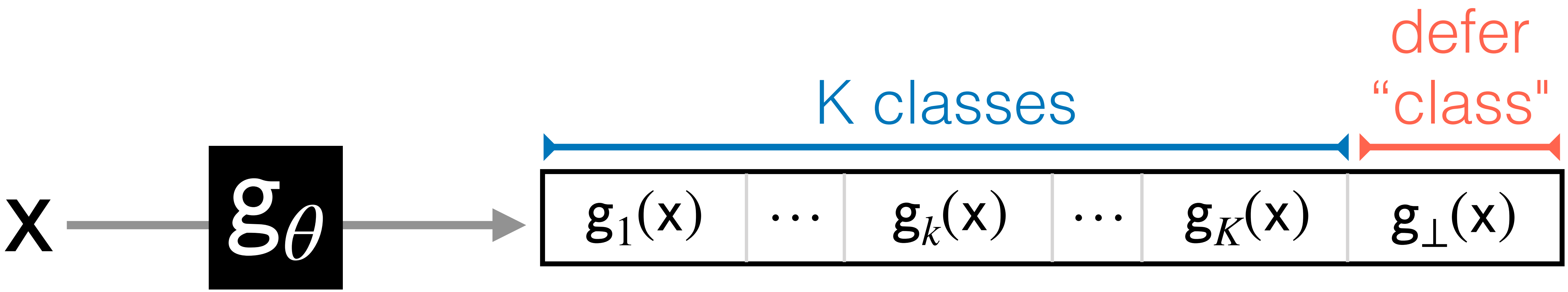
model

softmax implementation [Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n\}_{n=1}^{N}$$

model



softmax implementation

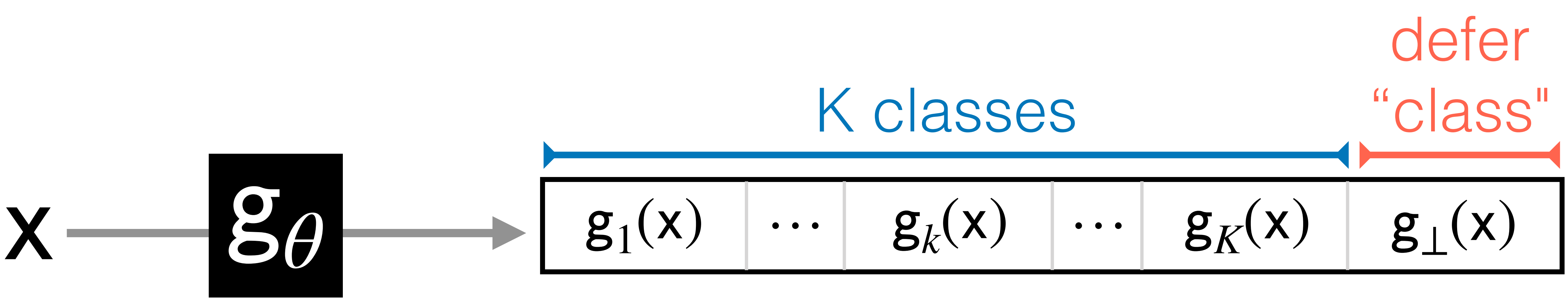
[Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n\}_{n=1}^{N}$$

model

$$\mathbf{g}_k(\mathrm{x}) \in \mathbb{R}$$



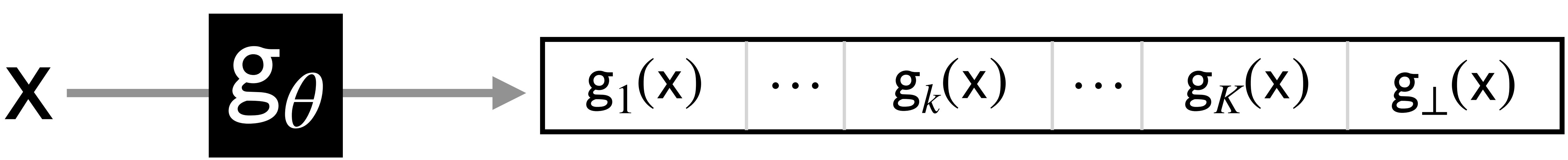
softmax implementation

[Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n\}_{n=1}^{N}$$

model



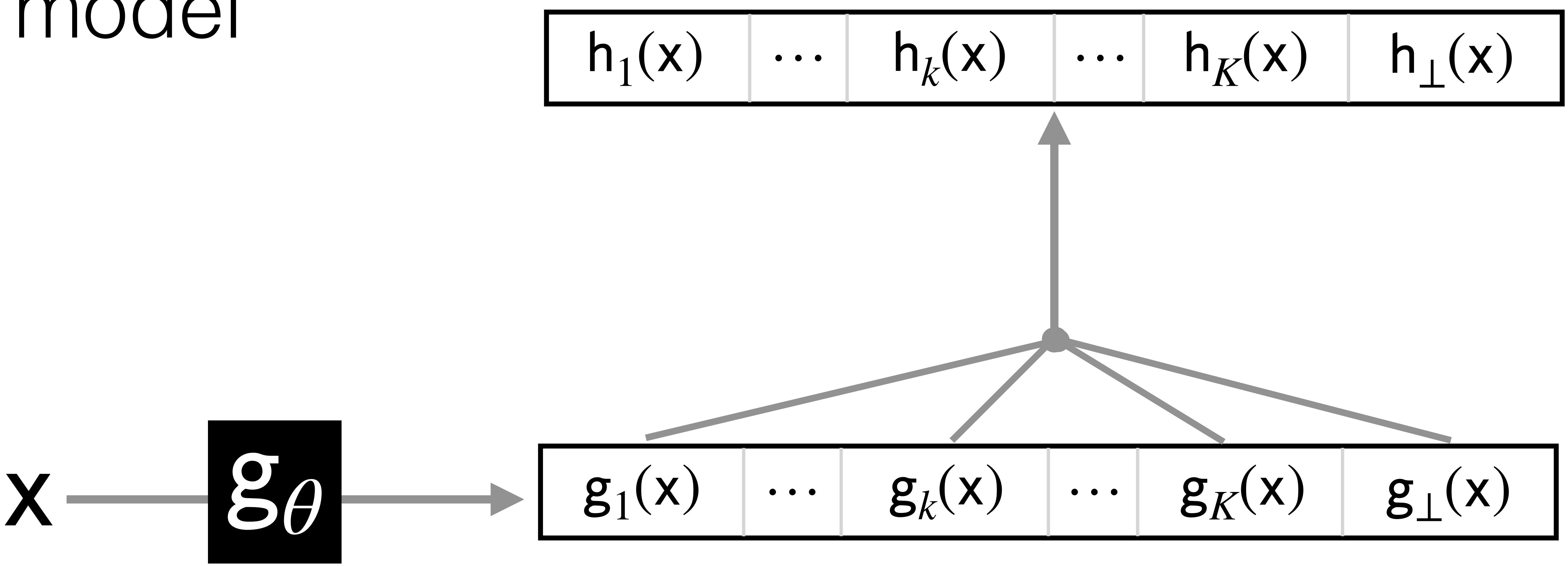
softmax implementation

[Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n\}_{n=1}^{N}$$

model



softmax implementation

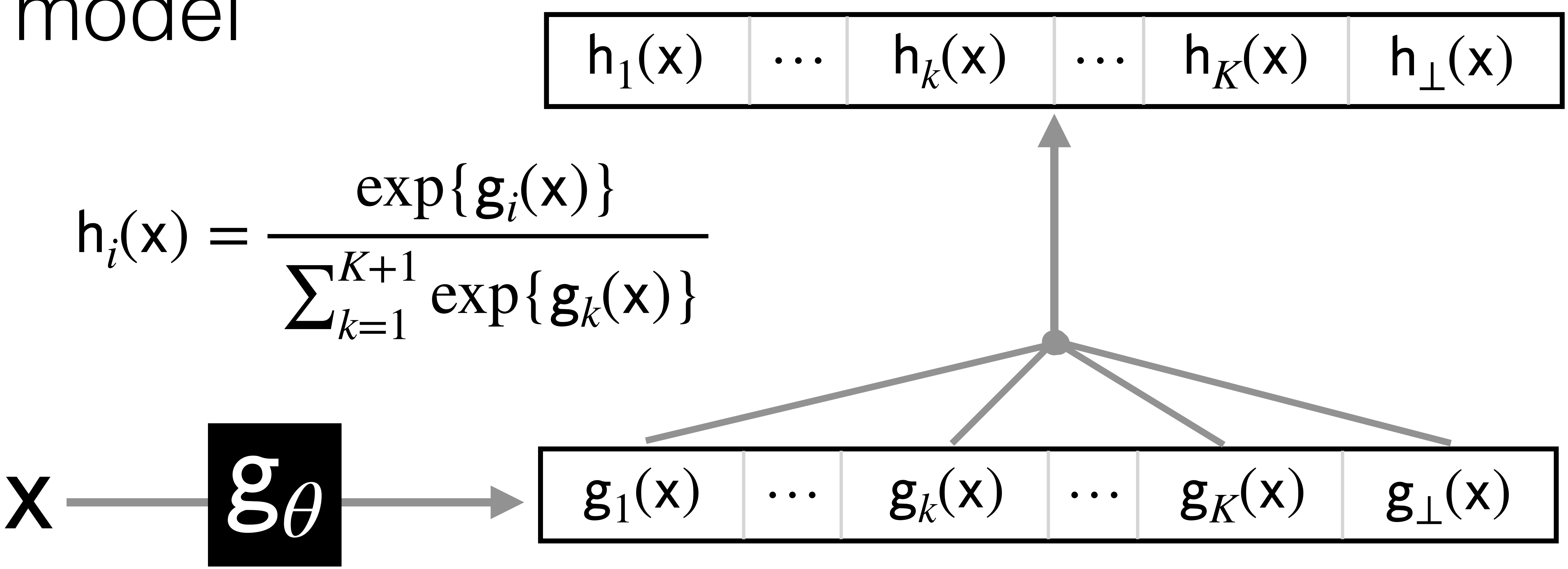
[Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathsf{x}_n, \mathsf{y}_n, \mathsf{m}_n\}_{n=1}^{N}$$

model

$$\mathsf{h}_i(\mathsf{x}) = \frac{\exp\{\mathsf{g}_i(\mathsf{x})\}}{\sum_{k=1}^{K+1} \exp\{\mathsf{g}_k(\mathsf{x})\}}$$



softmax implementation

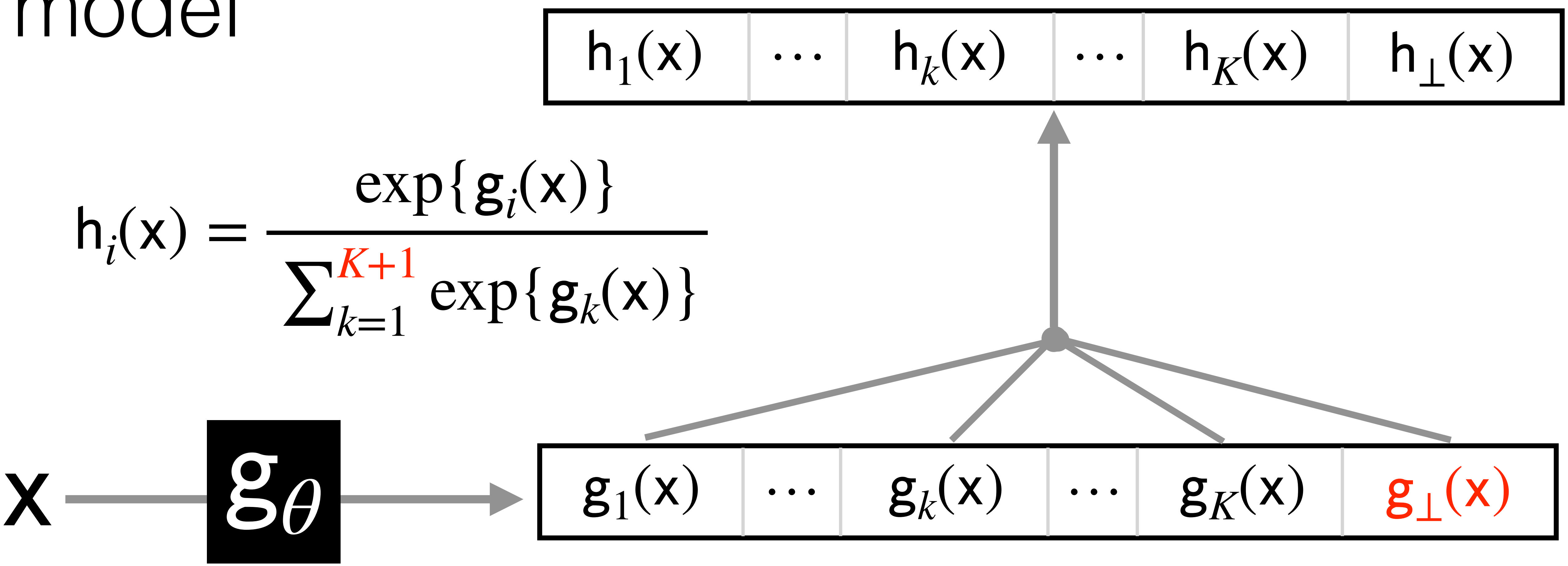
[Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n\}_{n=1}^{N}$$

model

$$\mathrm{h}_i(\mathrm{x}) = \frac{\exp\{\mathrm{g}_i(\mathrm{x})\}}{\sum_{k=1}^{K+1} \exp\{\mathrm{g}_k(\mathrm{x})\}}$$



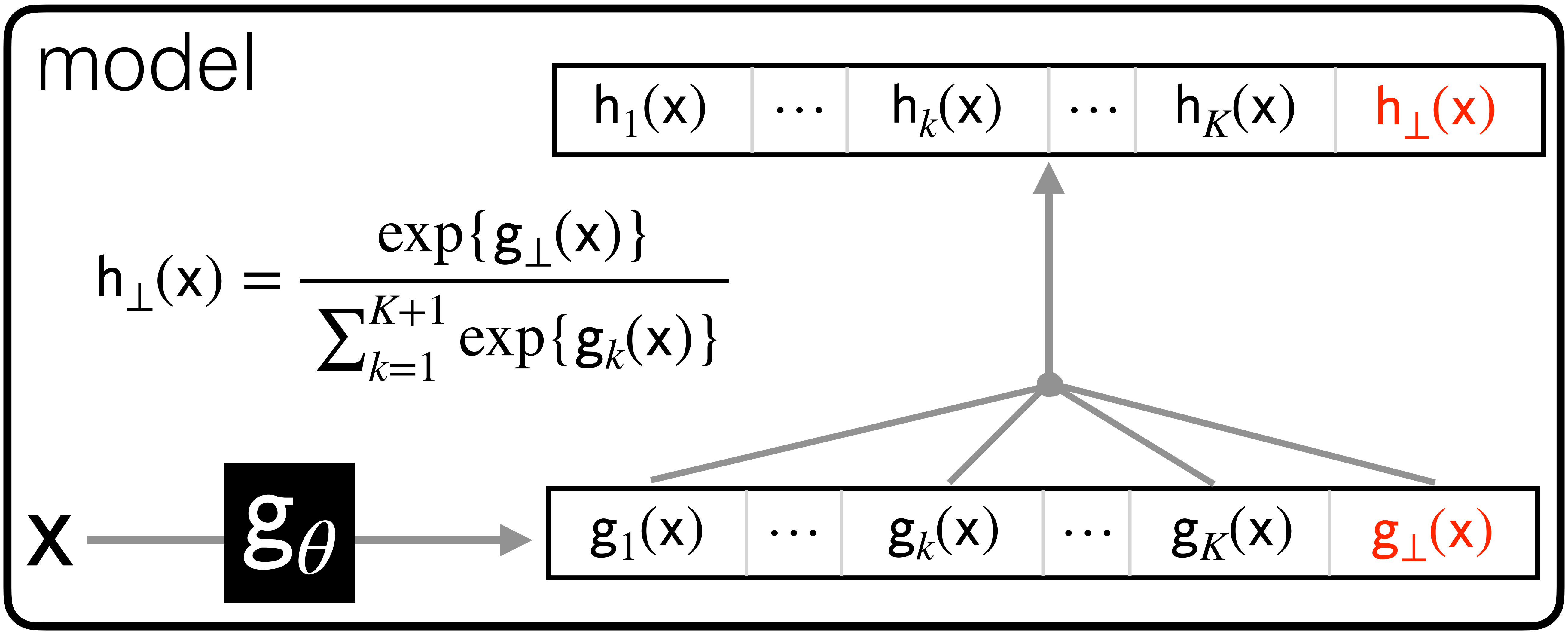
softmax implementation

[Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n\}_{n=1}^{N}$$

model



$$\mathrm{h}_\perp(\mathrm{x}) = \frac{\exp\{\mathrm{g}_\perp(\mathrm{x})\}}{\sum_{k=1}^{K+1} \exp\{\mathrm{g}_k(\mathrm{x})\}}$$

softmax implementation [Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \left\{ \mathsf{x}_n, \mathsf{y}_n, \mathsf{m}_n \right\}_{n=1}^{N}$$

model

$$\mathsf{h}_i(\mathsf{x}) = \frac{\exp\{\mathsf{g}_i(\mathsf{x})\}}{\sum_{k=1}^{K+1} \exp\{\mathsf{g}_k(\mathsf{x})\}}$$

softmax implementation

[Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n\}_{n=1}^{N}$$

model

$$\mathrm{h}_i(\mathrm{x}) = \frac{\exp\{\mathrm{g}_i(\mathrm{x})\}}{\sum_{k=1}^{K+1} \exp\{\mathrm{g}_k(\mathrm{x})\}}$$

loss function

$$\ell(\theta; \mathrm{x}, \mathrm{y}, \mathrm{m}) = -\log \mathrm{h}_\mathrm{y}(\mathrm{x}) \;-\; \mathbb{I}[\mathrm{y} = \mathrm{m}] \cdot \log \mathrm{h}_\perp(\mathrm{x})$$

softmax implementation

[Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n\}_{n=1}^{N}$$

model

$$\mathrm{h}_i(\mathrm{x}) = \frac{\exp\{\mathrm{g}_i(\mathrm{x})\}}{\sum_{k=1}^{K+1} \exp\{\mathrm{g}_k(\mathrm{x})\}}$$

loss function

$$\ell(\theta; \mathrm{x}, \mathrm{y}, \mathrm{m}) = -\log \mathrm{h}_{\mathrm{y}}(\mathrm{x}) \; - \; \mathbb{I}[\mathrm{y} = \mathrm{m}] \cdot \log \mathrm{h}_{\perp}(\mathrm{x})$$

softmax implementation

[Mozannar & Sontag, 2020]

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n\}_{n=1}^{N}$$

model

$$\mathrm{h}_i(\mathrm{x}) = \frac{\exp\{\mathrm{g}_i(\mathrm{x})\}}{\sum_{k=1}^{K+1} \exp\{\mathrm{g}_k(\mathrm{x})\}}$$

loss function

$$\ell(\theta; \mathrm{x}, \mathrm{y}, \mathrm{m}) = -\log \mathrm{h}_{\mathrm{y}}(\mathrm{x}) \; - \; \mathbb{I}[\mathrm{y} = \mathrm{m}] \cdot \log \mathrm{h}_{\perp}(\mathrm{x})$$

softmax implementation [Mozannar & Sontag, 2020]

training data

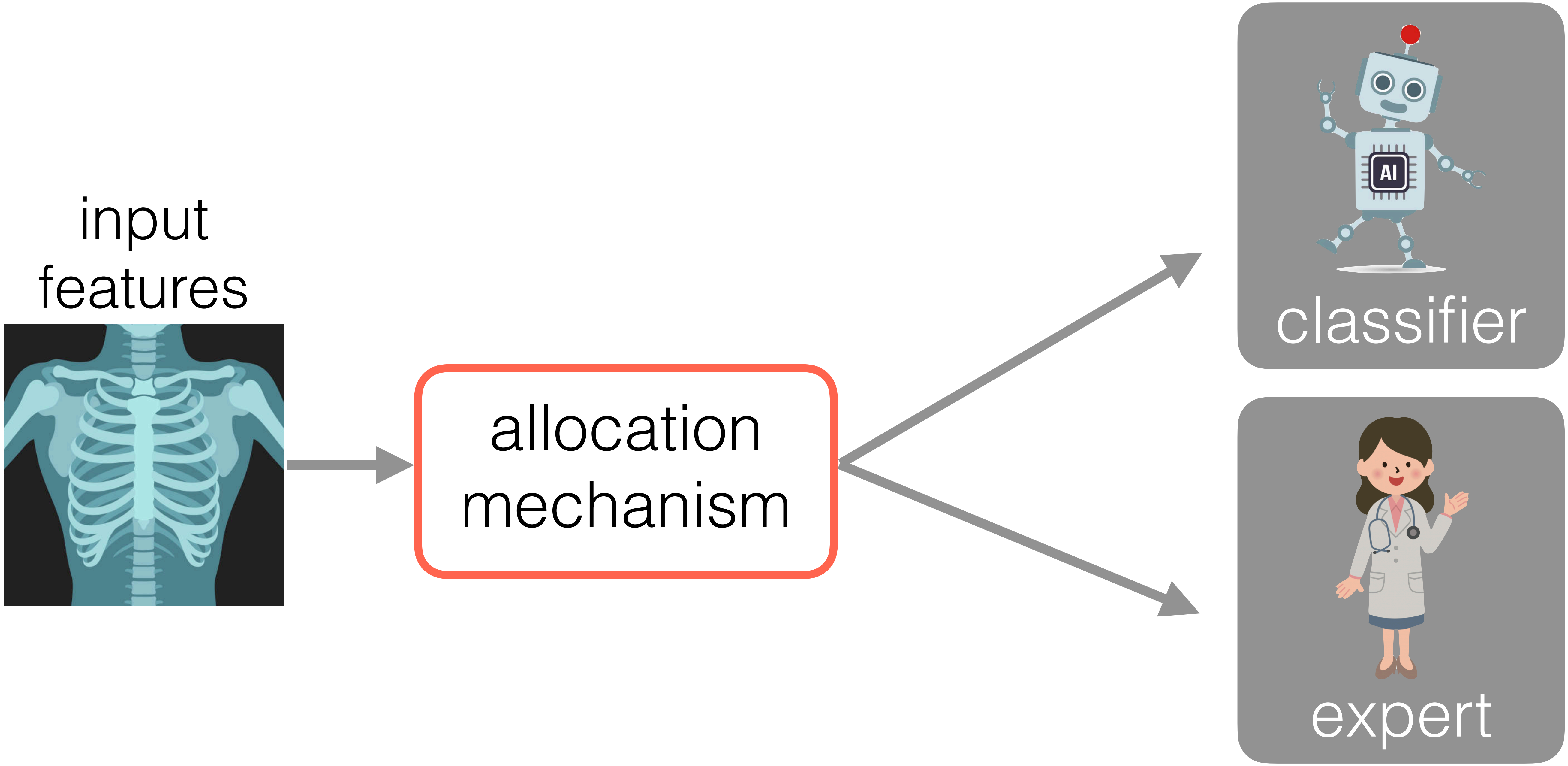
$$\mathfrak{D} = \left\{ \mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_n \right\}_{n=1}^{N}$$

model

$$\mathrm{h}_i(\mathrm{x}) = \frac{\exp\{\mathbf{g}_i(\mathrm{x})\}}{\sum_{k=1}^{K+1} \exp\{\mathbf{g}_k(\mathrm{x})\}}$$

loss function

$$\ell(\theta; \mathrm{x}, \mathrm{y}, \mathrm{m}) = -\log \mathrm{h}_{\mathrm{y}}(\mathrm{x}) \;-\; \mathbb{I}[\mathrm{y} = \mathrm{m}] \cdot \log \mathrm{h}_{\perp}(\mathrm{x})$$



defer to expert if…

$$\max_{y \in [1,K]} \; \mathrm{h}_y(\mathrm{x}) \leq \mathrm{h}_{\perp}(\mathrm{x})$$

- ⊗ single expert
  - ⊗ softmax surrogate loss
  - ⊗ improving calibration via one-vs-all
- ⊗ multiple experts
  - ⊗ surrogate losses
  - ⊗ conformal sets of experts
- ⊗ population of experts
  - ⊗ surrogate losses
  - ⊗ meta-learning a rejector

- ⊗ single expert
  - ⊗ softmax surrogate loss
  - ⊗ improving calibration via one-vs-all
- ⊗ multiple experts
  - ⊗ surrogate losses
  - ⊗ conformal sets of experts
- ⊗ population of experts
  - ⊗ surrogate losses
  - ⊗ meta-learning a rejector

How well can the softmax-based system estimate expert correctness?

$$\hat{p}(m = y \mid x) \overset{?}{\approx} \mathbb{P}(m = y \mid x)$$

How well can the softmax-based system estimate expert correctness?

$$\hat{p}(m = y \mid x) \underset{?}{\approx} \mathbb{P}(m = y \mid x)$$

- ⊗ optimal allocation
- ⊗ transparency
- ⊗ detecting distribution shift (in the expert)

How well can the softmax-based system estimate expert correctness?

$$\hat{p}(m = y \mid x) \not\approx \mathbb{P}(m = y \mid x)$$

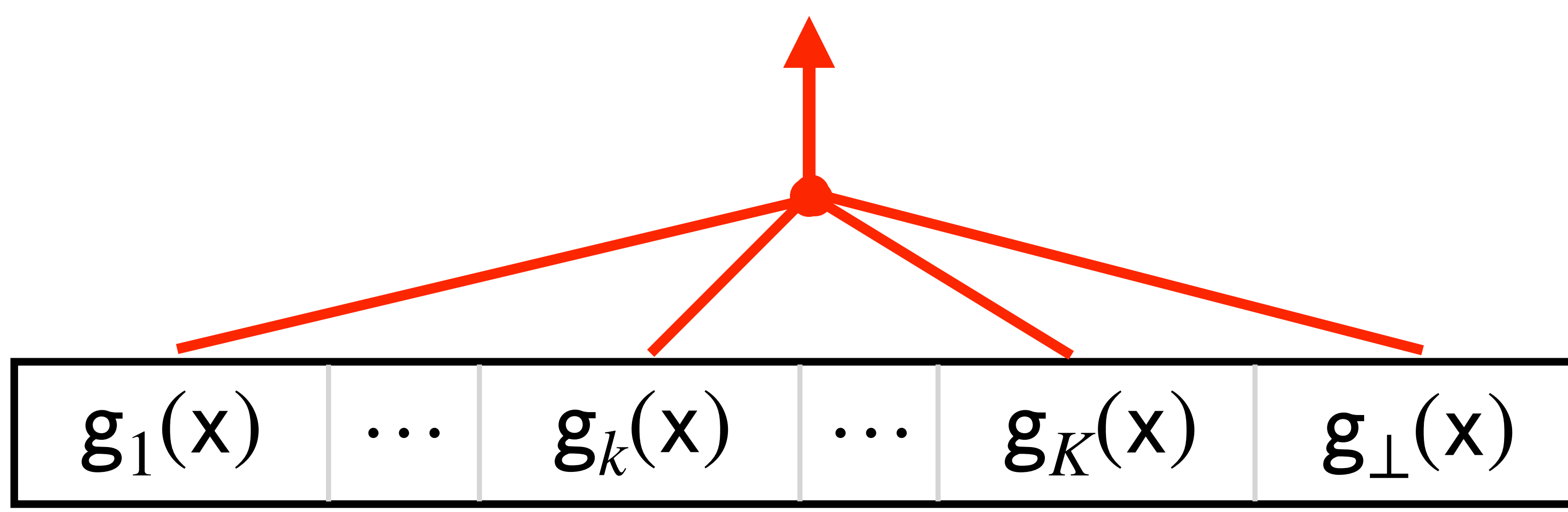
How well can the softmax-based system estimate expert correctness?

$$\hat{p}(m = y \mid x) \not\approx \mathbb{P}(m = y \mid x)$$

degenerate parameterization

[Proposition 3.1]

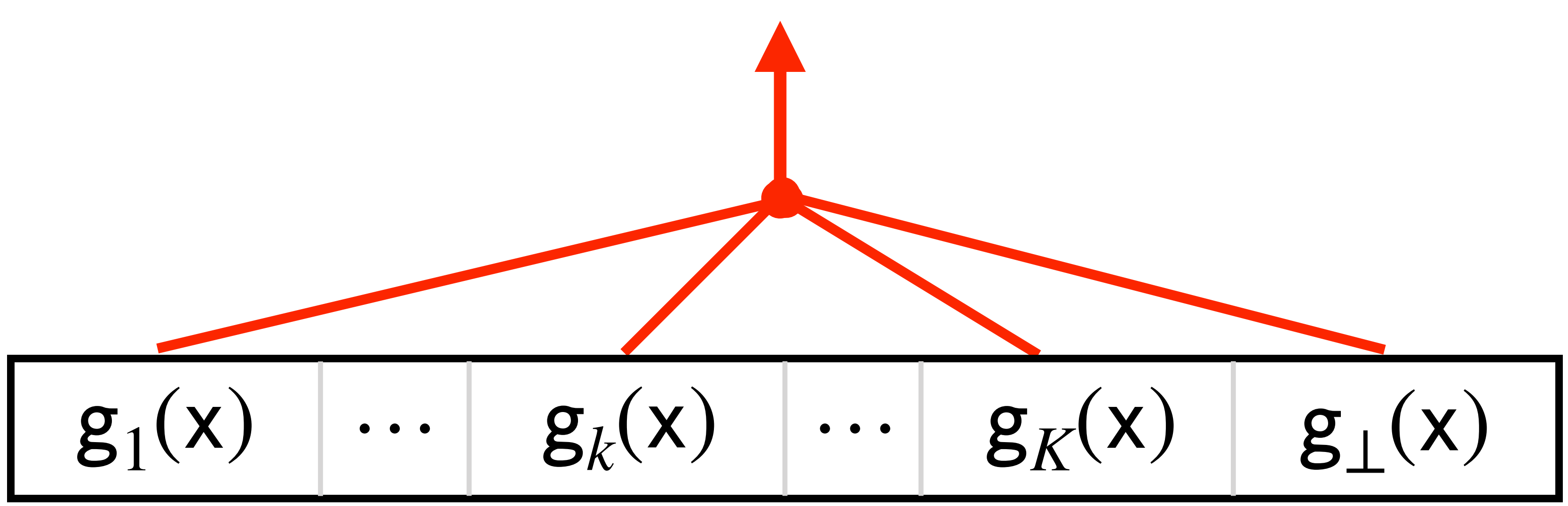
$$h_{\perp}(x) = \frac{\exp\{g_{\perp}(x)\}}{\sum_{k=1}^{K+1} \exp\{g_k(x)\}}$$



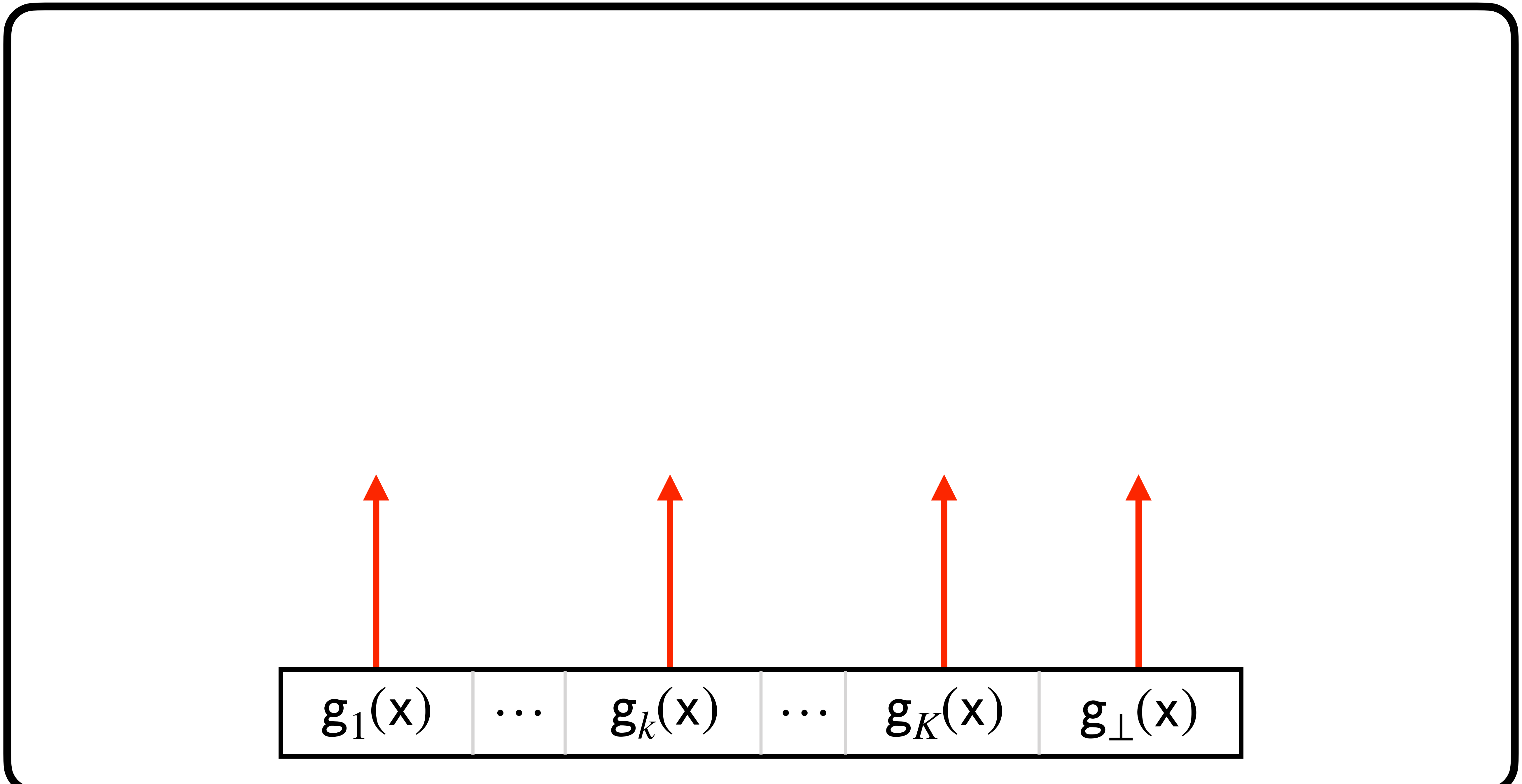
our solution: switch to a **one-vs-all** parameterization

our solution: switch to a **one-vs-all** parameterization

$$h_{\perp}(x) = \frac{\exp\{g_{\perp}(x)\}}{\sum_{k=1}^{K+1} \exp\{g_k(x)\}}$$

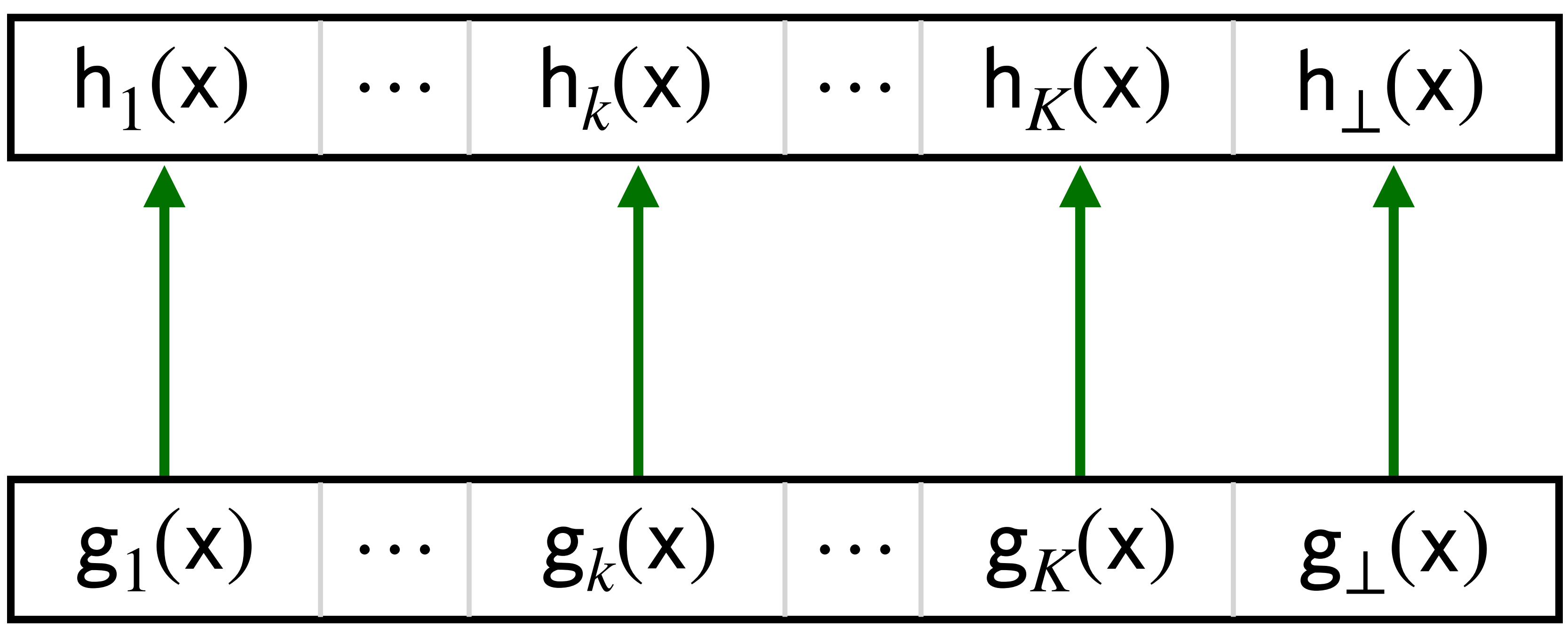


our solution: switch to a **one-vs-all** parameterization



our solution: switch to a **one-vs-all** parameterization

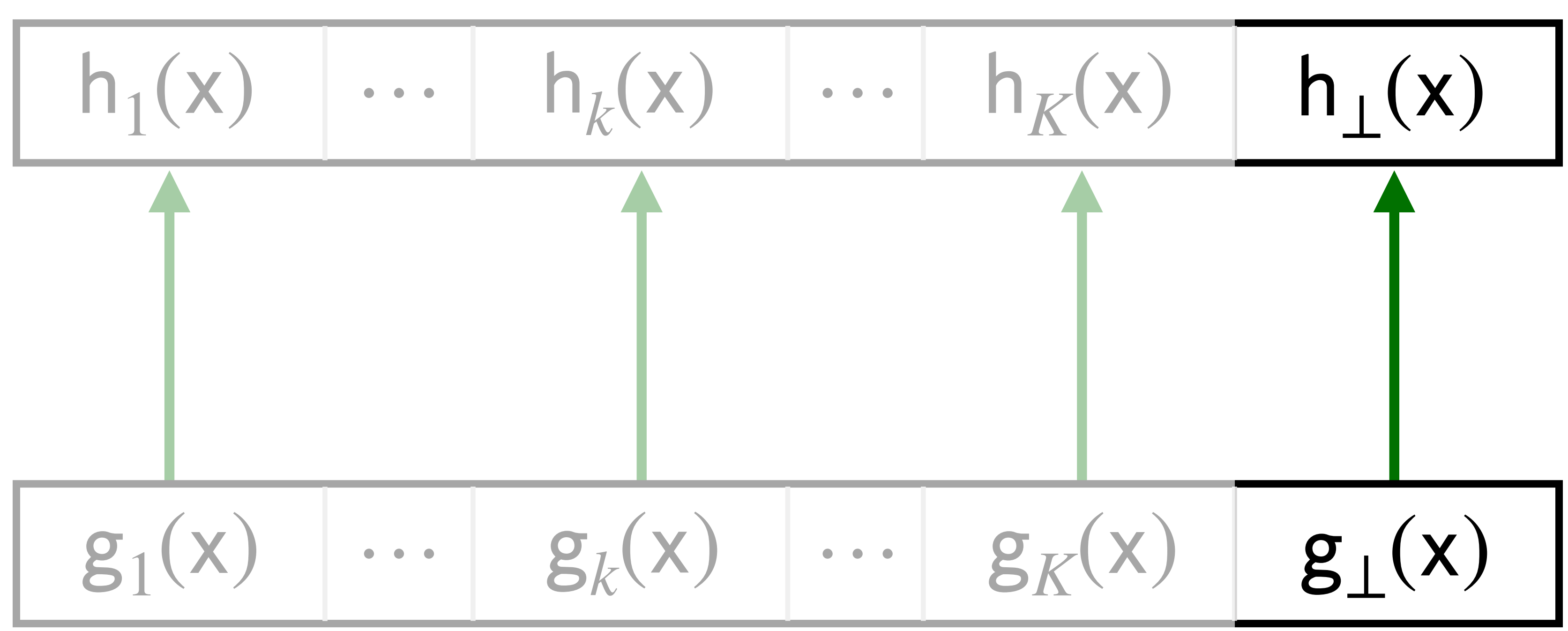
$$h_i(x) = \frac{1}{1 + \exp\{-g_i(x)\}}$$



our solution: switch to a **one-vs-all** parameterization

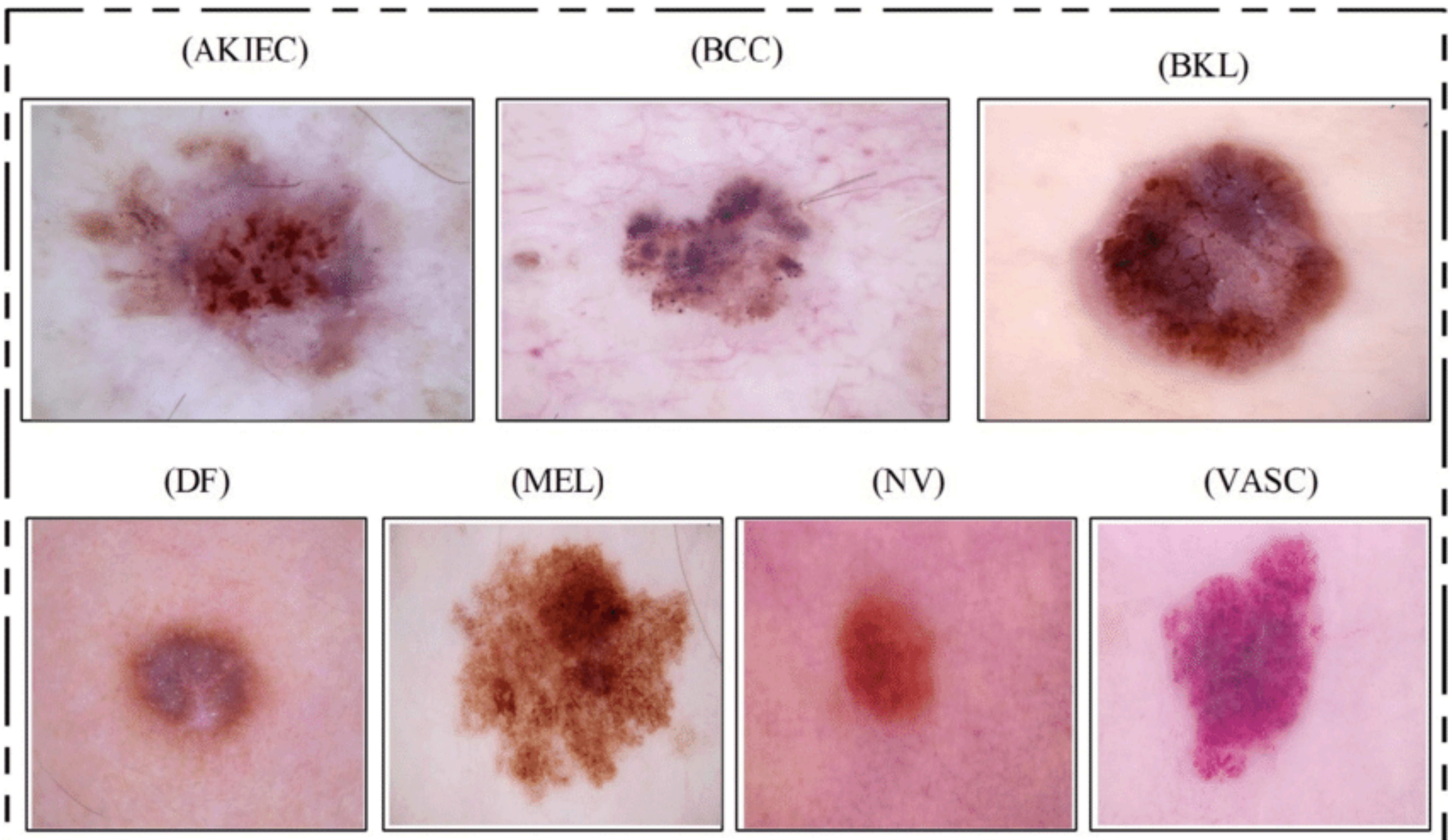
[Theorem 4.1]

$$h^*_\perp(x) = \mathbb{P}(m = y \mid x)$$



estimating expert correctness

skin lesion diagnosis



estimating expert correctness

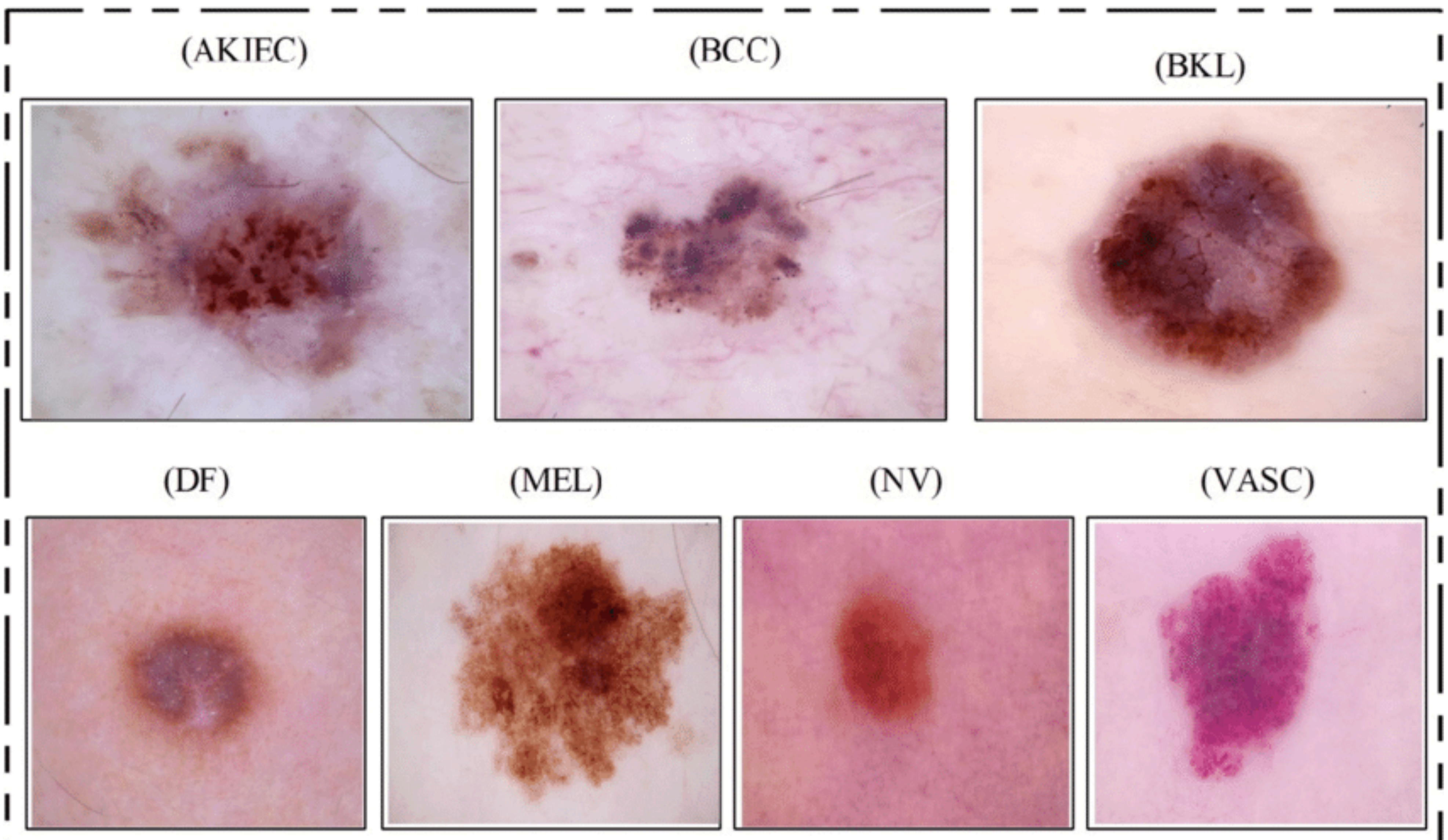
$\hat{p}$	distance: $\hat{p}$ vs $\mathbb{P}$
softmax	
one-vs-all (ours)	

estimating expert correctness

$\hat{p}$	distance: $\hat{p}$ vs $\mathbb{P}$
softmax	26.7 ± 1.8
one-vs-all (ours)	8.0 ± 1.0 ✓

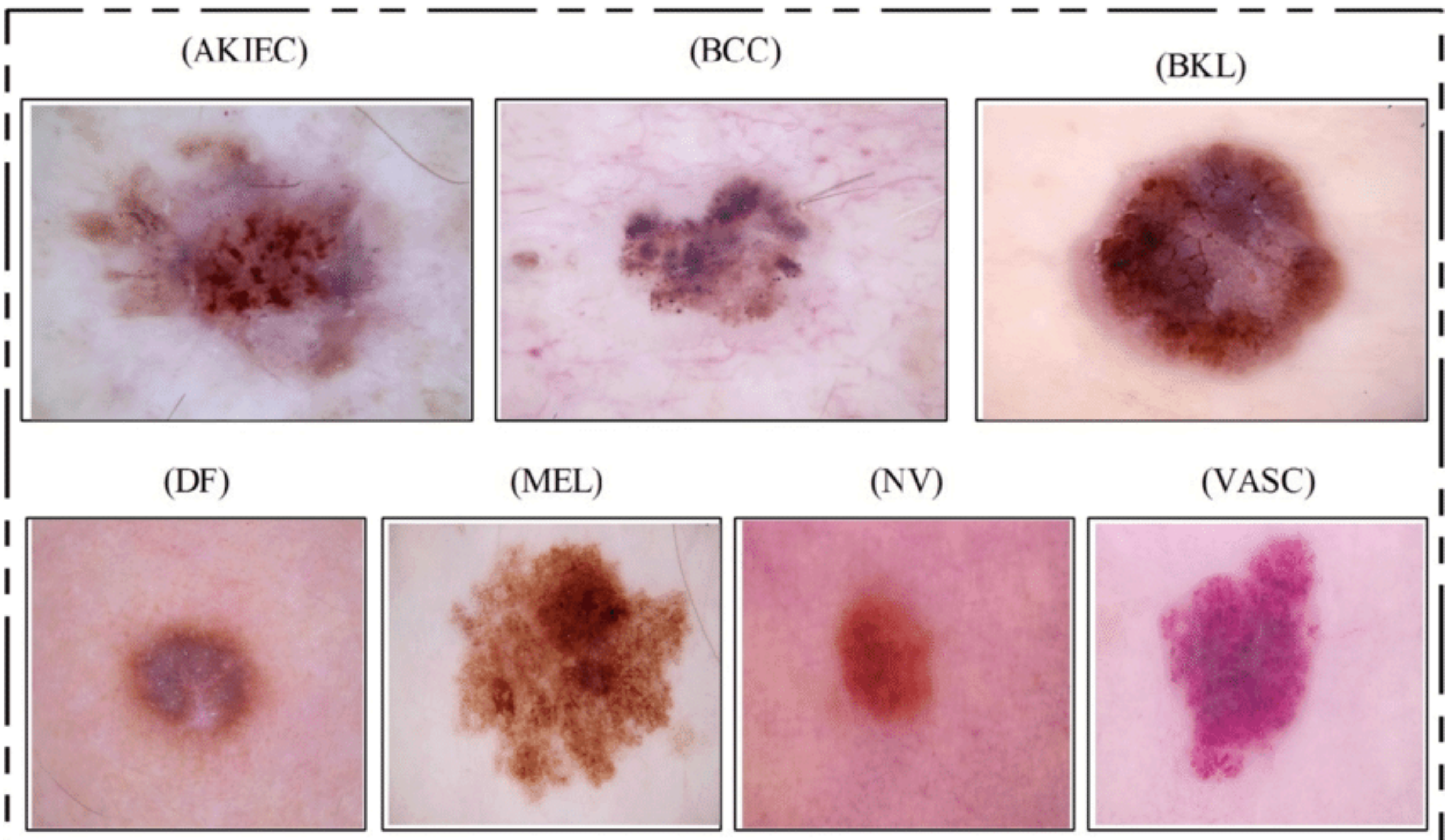
But does one-vs-all
result in **more accurate** models?

skin lesion diagnosis



one-vs-all (ours)   softmax

skin lesion diagnosis



one-vs-all (ours)

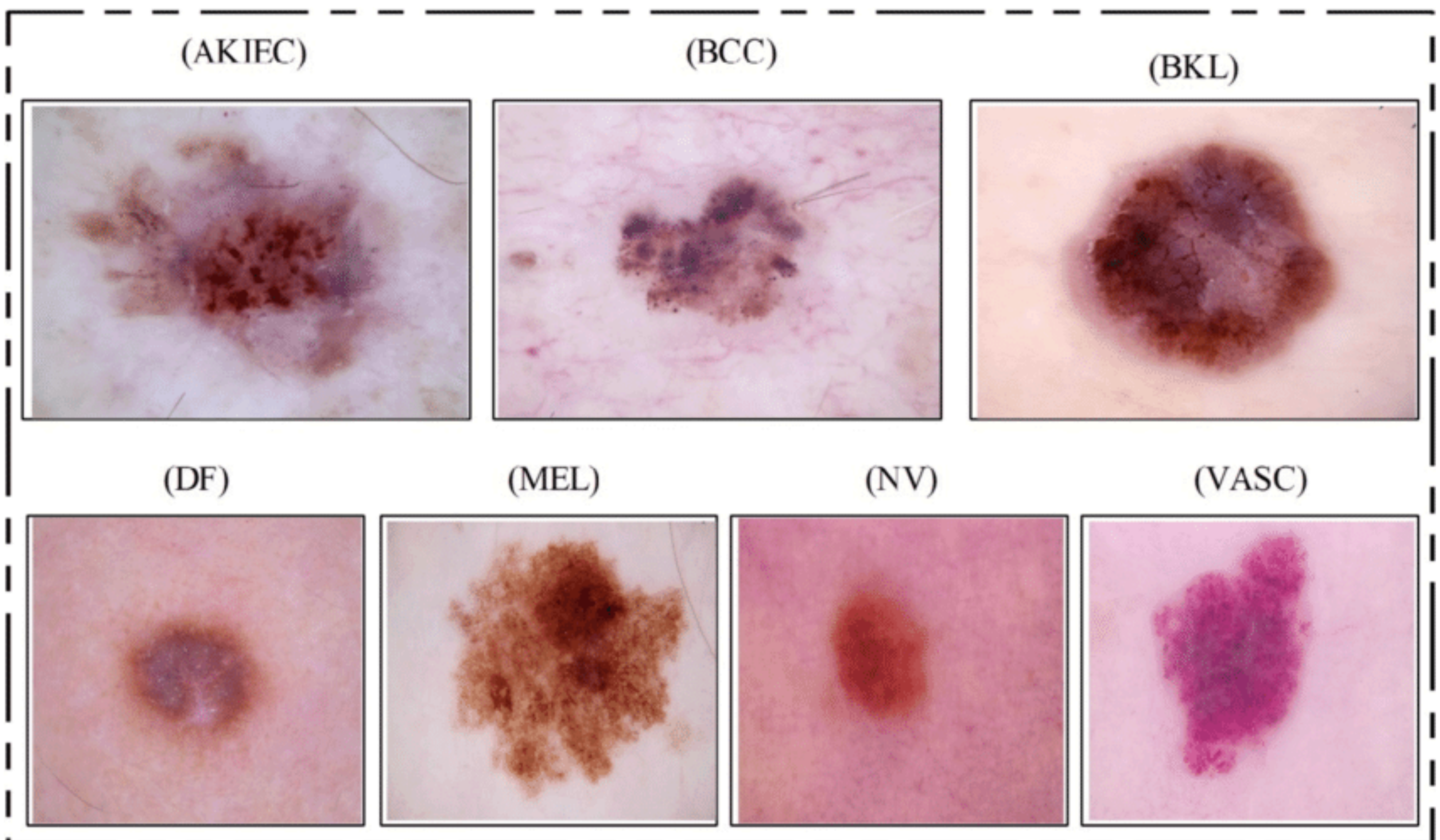
softmax

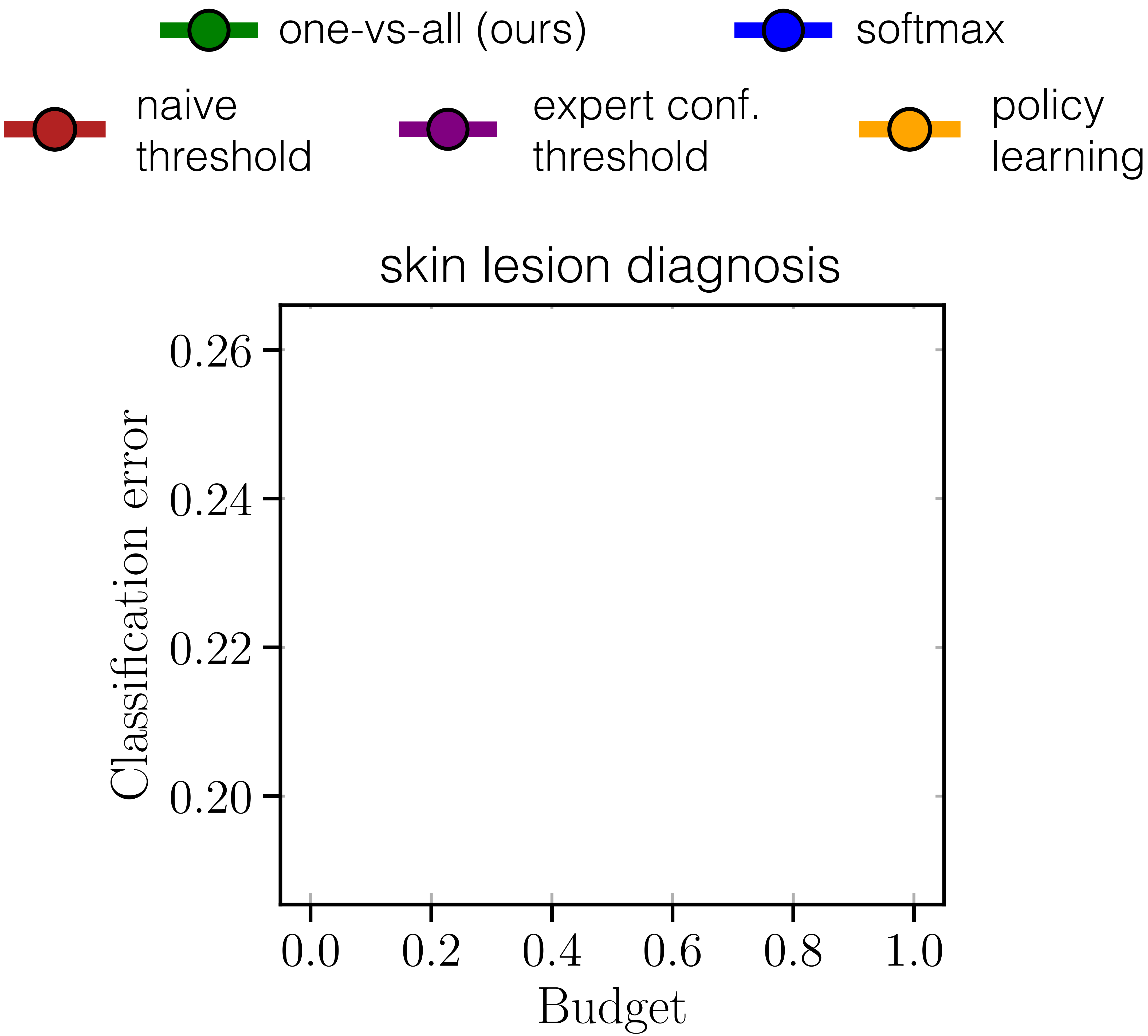
naive threshold

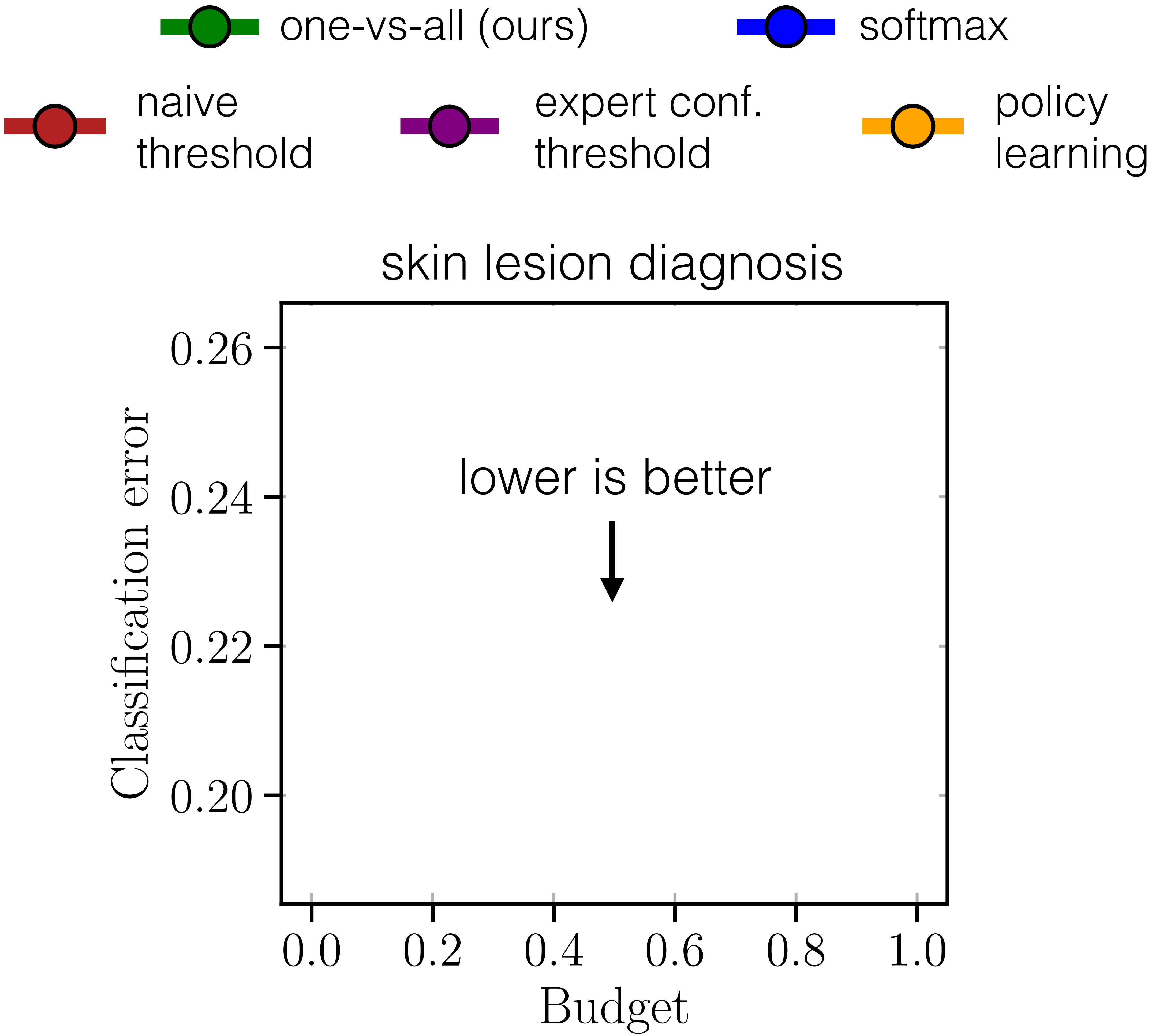
expert conf. threshold

policy learning

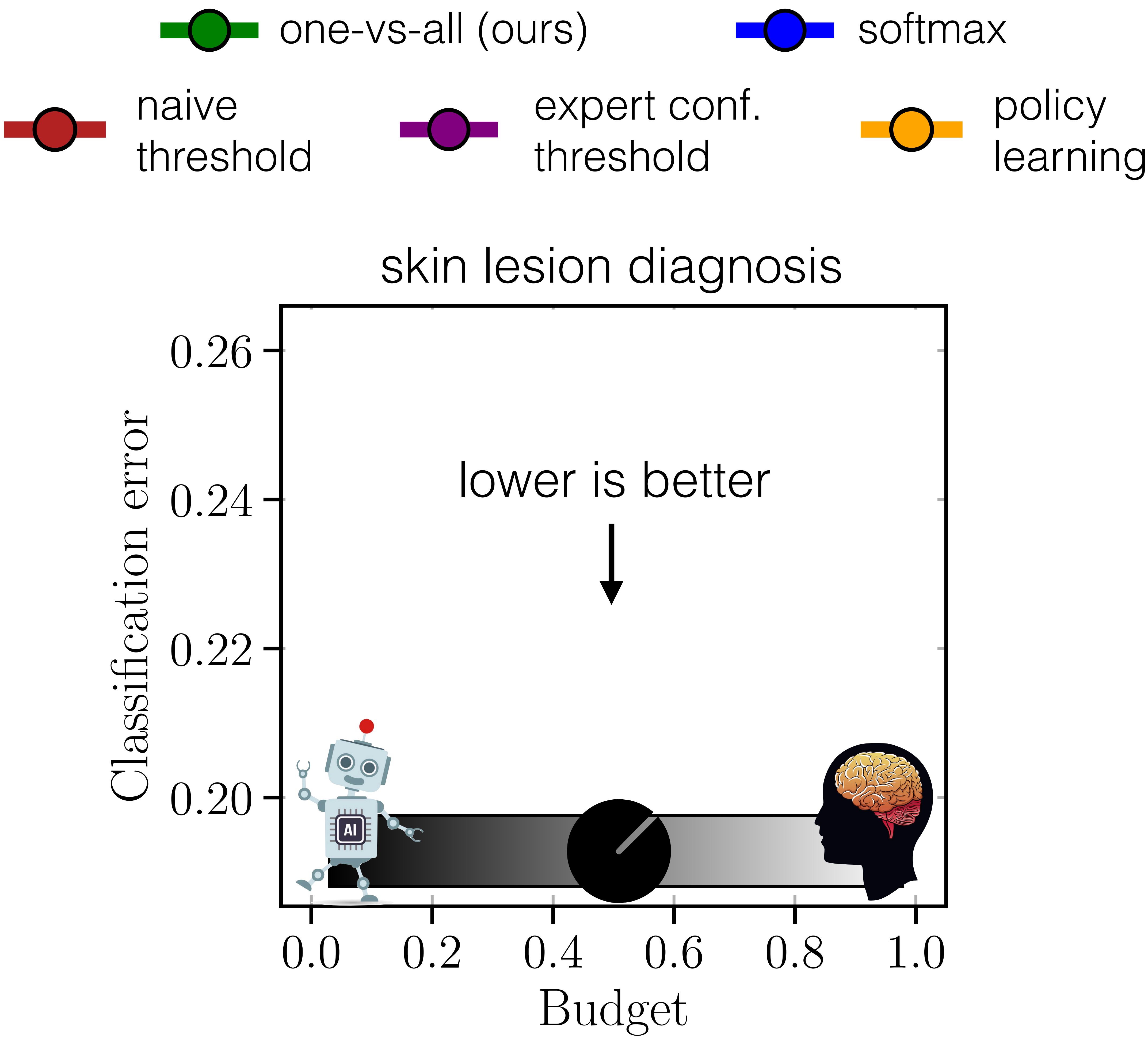
skin lesion diagnosis



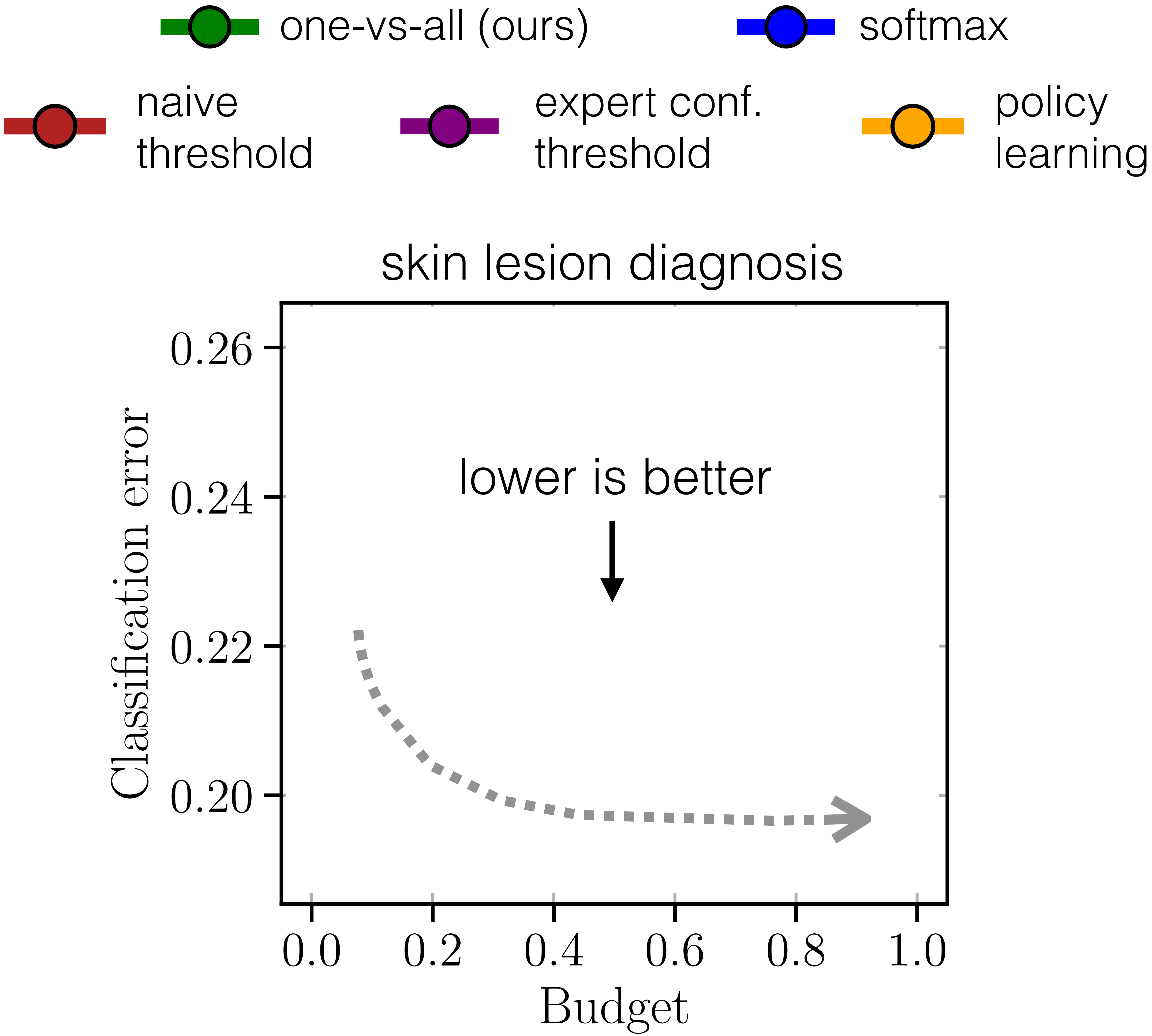

one-vs-all (ours)
softmax
naive threshold
expert conf. threshold
policy learning
skin lesion diagnosis
Classification error
0.20
0.22
0.24
0.26
0.0
0.2
0.4
0.6
0.8
1.0
Budget

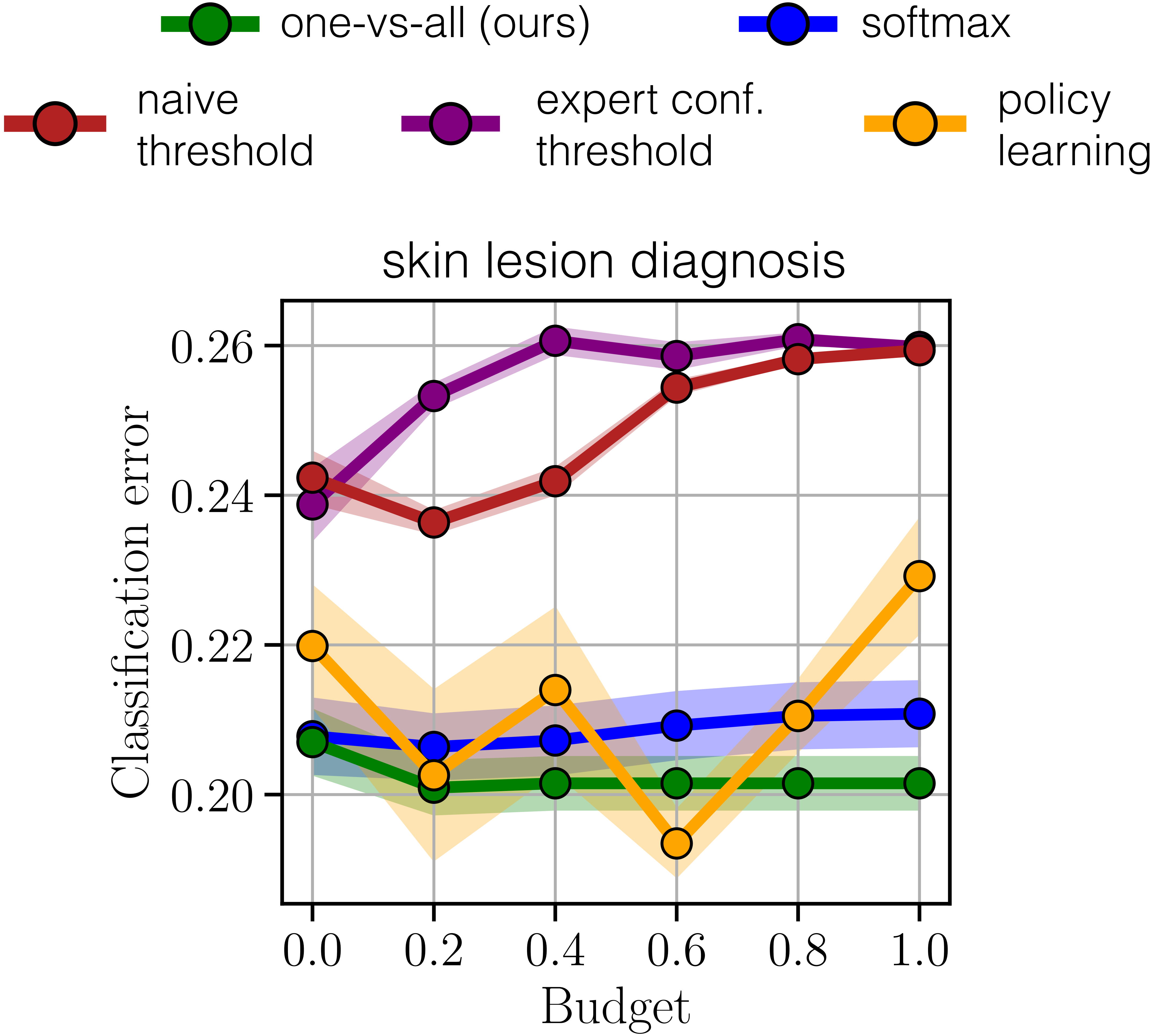


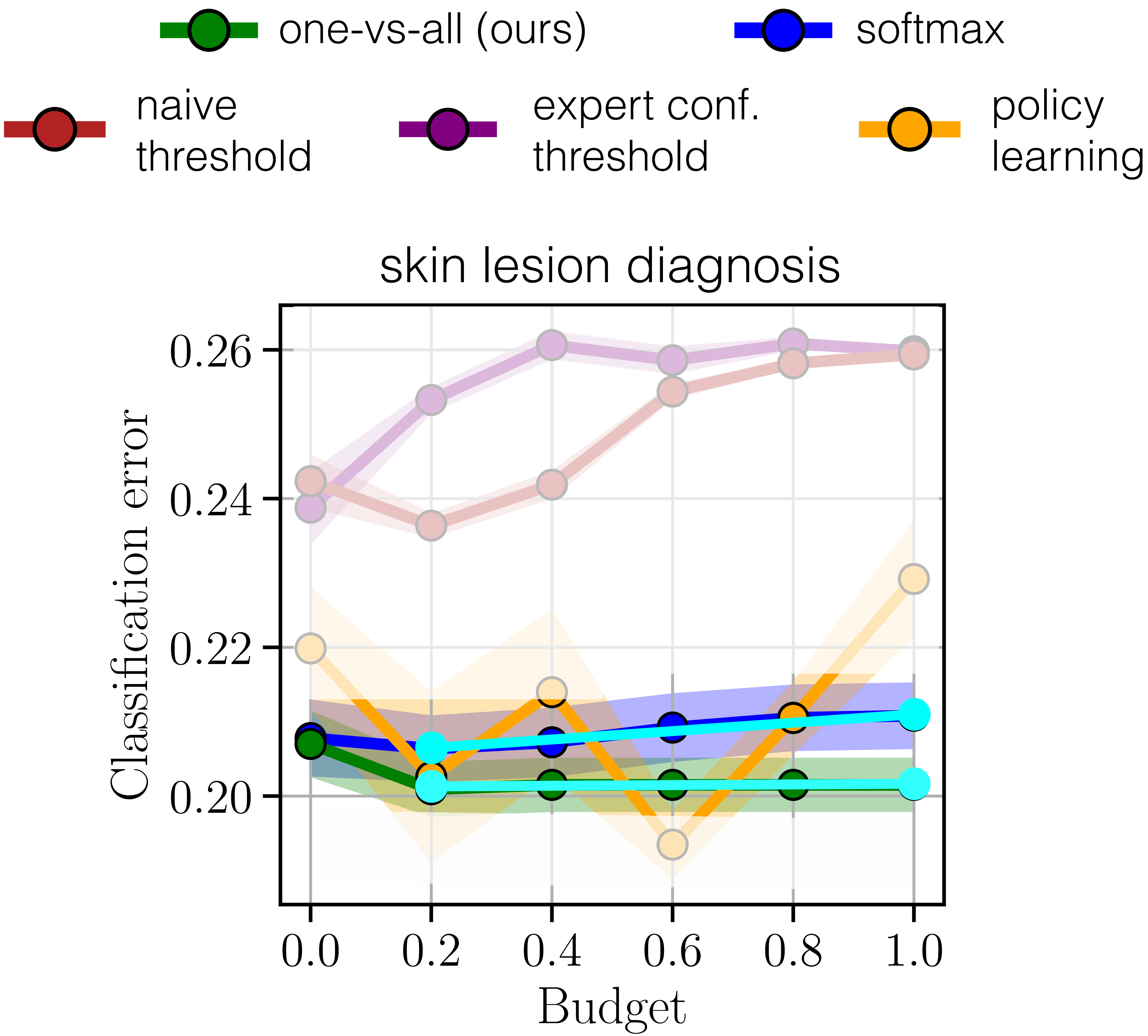
one-vs-all (ours)
softmax
naive threshold
expert conf. threshold
policy learning
skin lesion diagnosis
Classification error
0.26
0.24
0.22
0.20
lower is better
0.0
0.2
0.4
0.6
0.8
1.0
Budget



one-vs-all (ours)
softmax
naive threshold
expert conf. threshold
policy learning
skin lesion diagnosis
lower is better
AI
Classification error
0.26
0.24
0.22
0.20
0.0
0.2
0.4
0.6
0.8
1.0
Budget

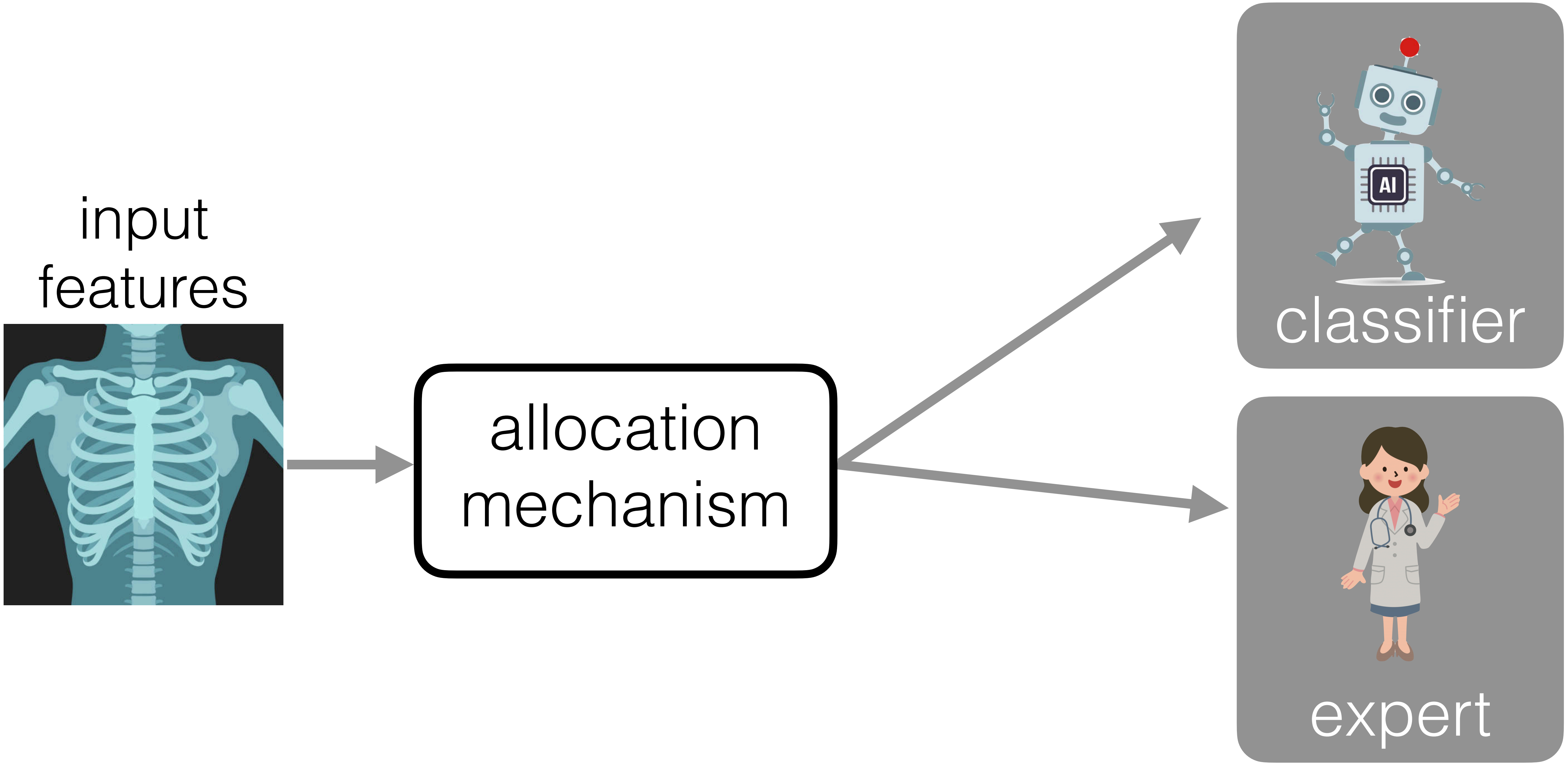

one-vs-all (ours)
softmax
naive threshold
expert conf. threshold
policy learning
skin lesion diagnosis
lower is better
Classification error
0.26
0.24
0.22
0.20
0.0
0.2
0.4
0.6
0.8
1.0
Budget


one-vs-all (ours)
softmax
naive threshold
expert conf. threshold
policy learning
skin lesion diagnosis
Classification error
0.26
0.24
0.22
0.20
0.0
0.2
0.4
0.6
0.8
1.0
Budget

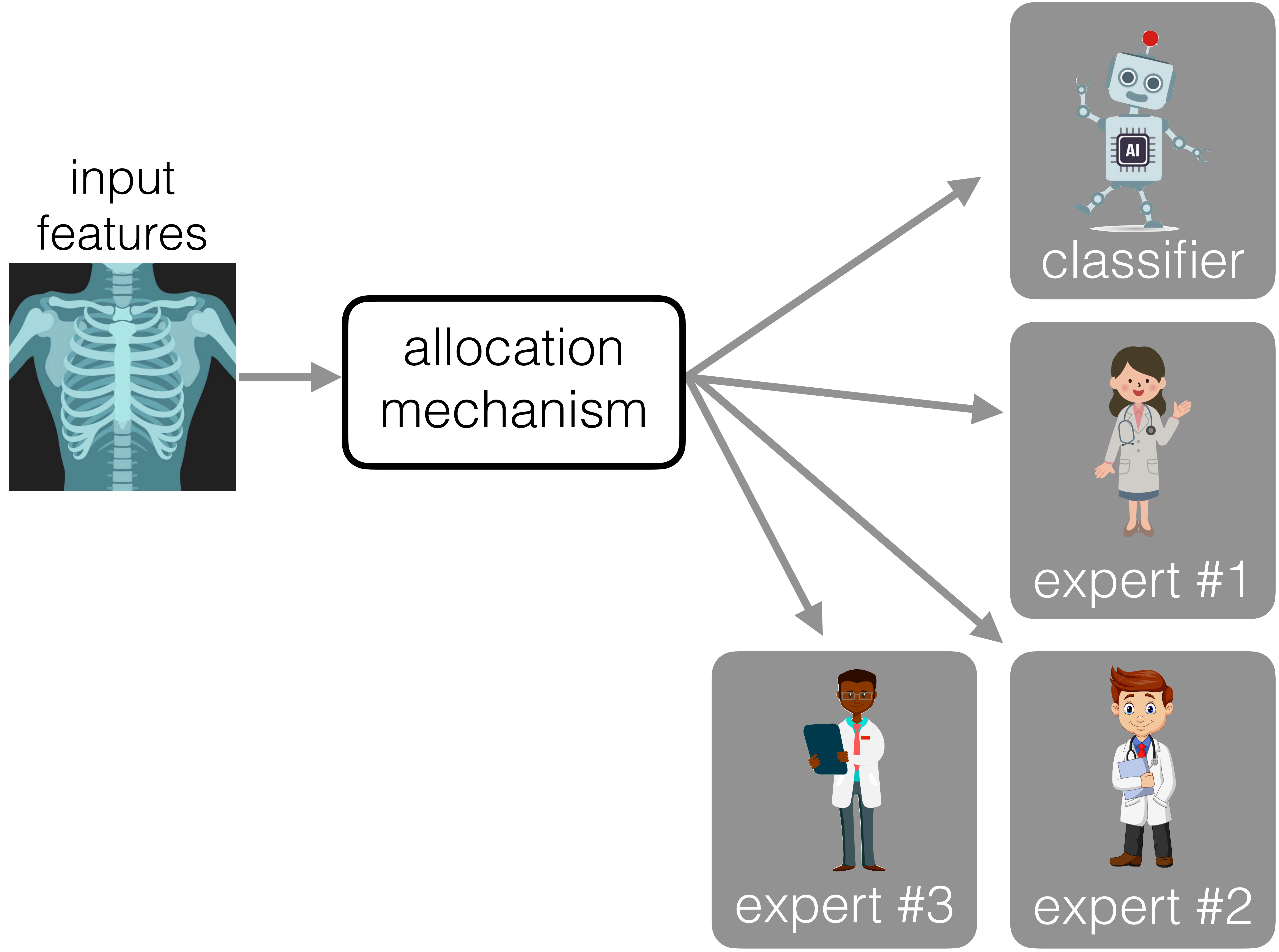

one-vs-all (ours)
softmax
naive threshold
expert conf. threshold
policy learning
skin lesion diagnosis
Classification error
0.26
0.24
0.22
0.20
0.0
0.2
0.4
0.6
0.8
1.0
Budget

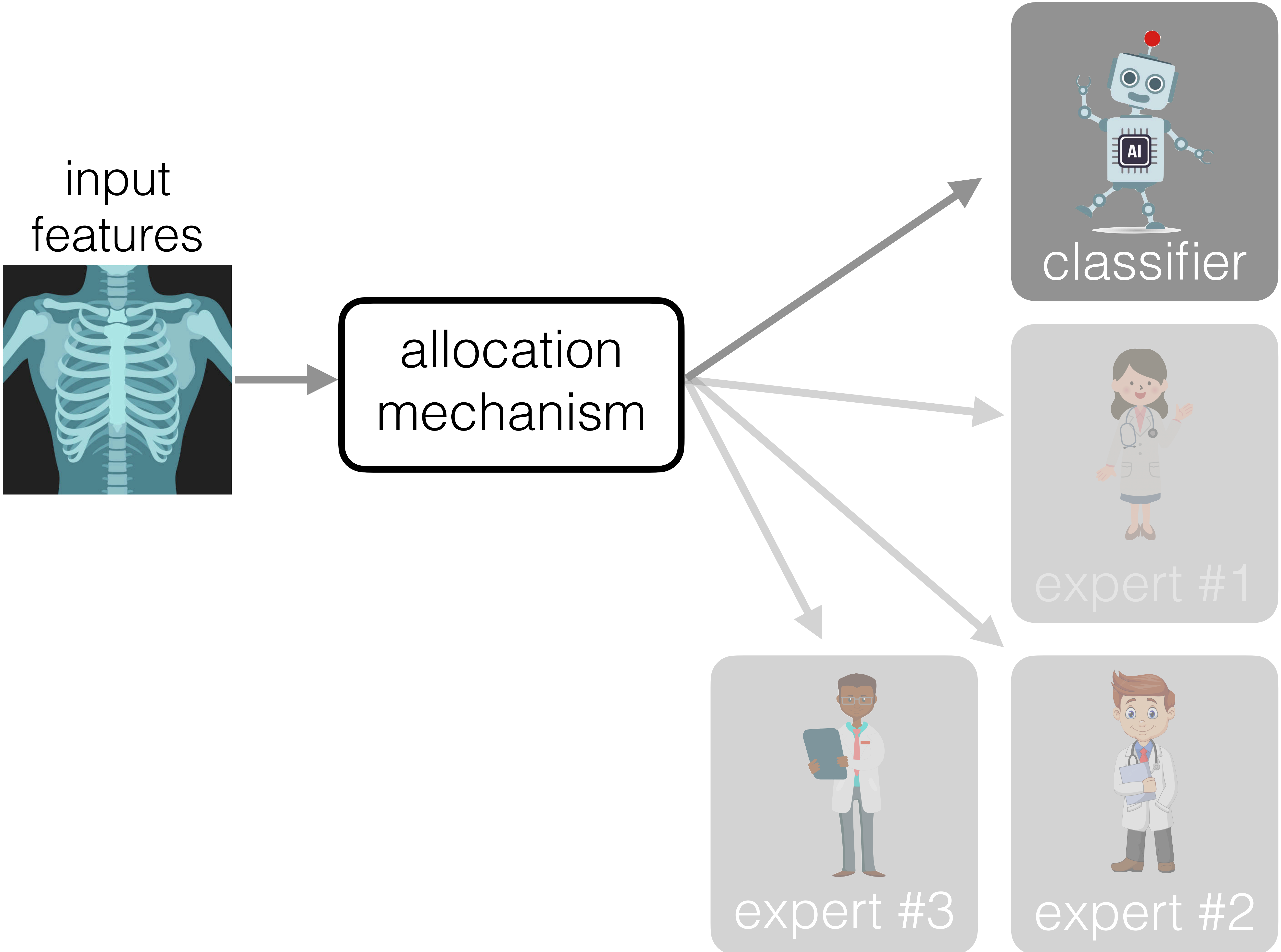
- ⊗ single expert
  - ⊗ softmax surrogate loss
  - ⊗ improving calibration via one-vs-all
- ⊗ multiple experts
  - ⊗ surrogate losses
  - ⊗ conformal sets of experts
- ⊗ population of experts
  - ⊗ surrogate losses
  - ⊗ meta-learning a rejector

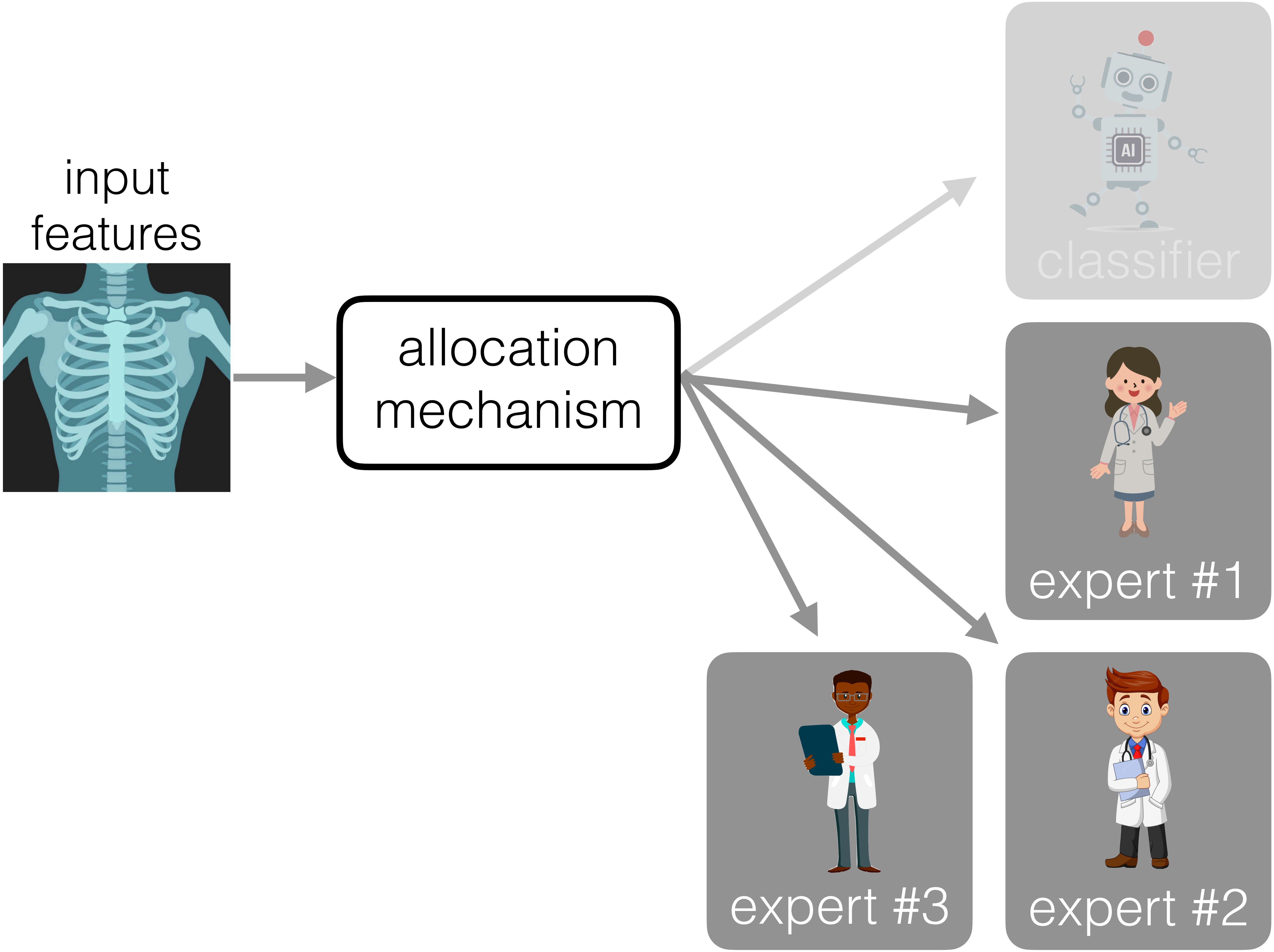
- single expert
  - softmax surrogate loss
  - improving calibration via one-vs-all
- **multiple experts**
  - surrogate losses
  - conformal sets of experts
- population of experts
  - surrogate losses
  - meta-learning a rejector

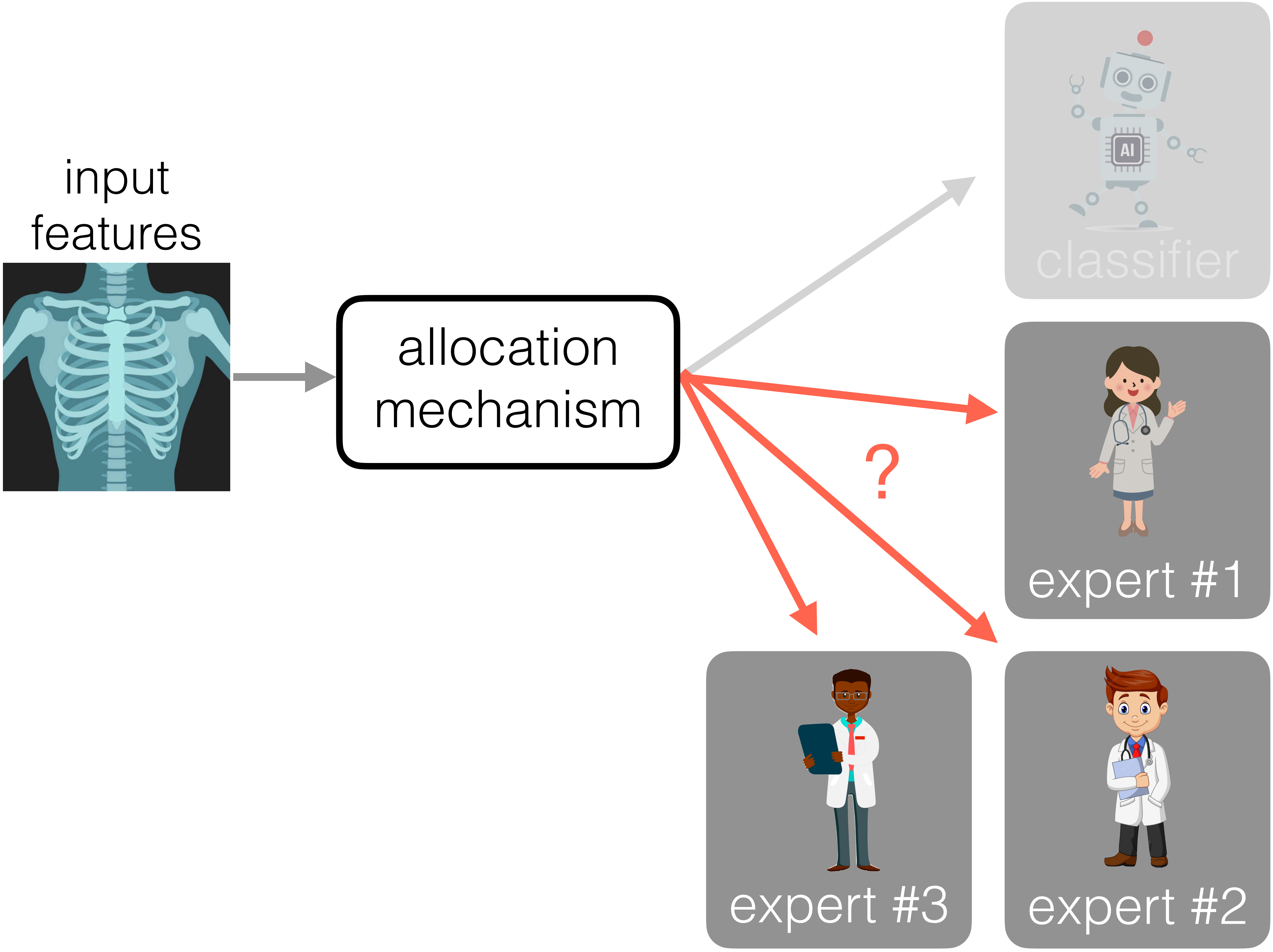


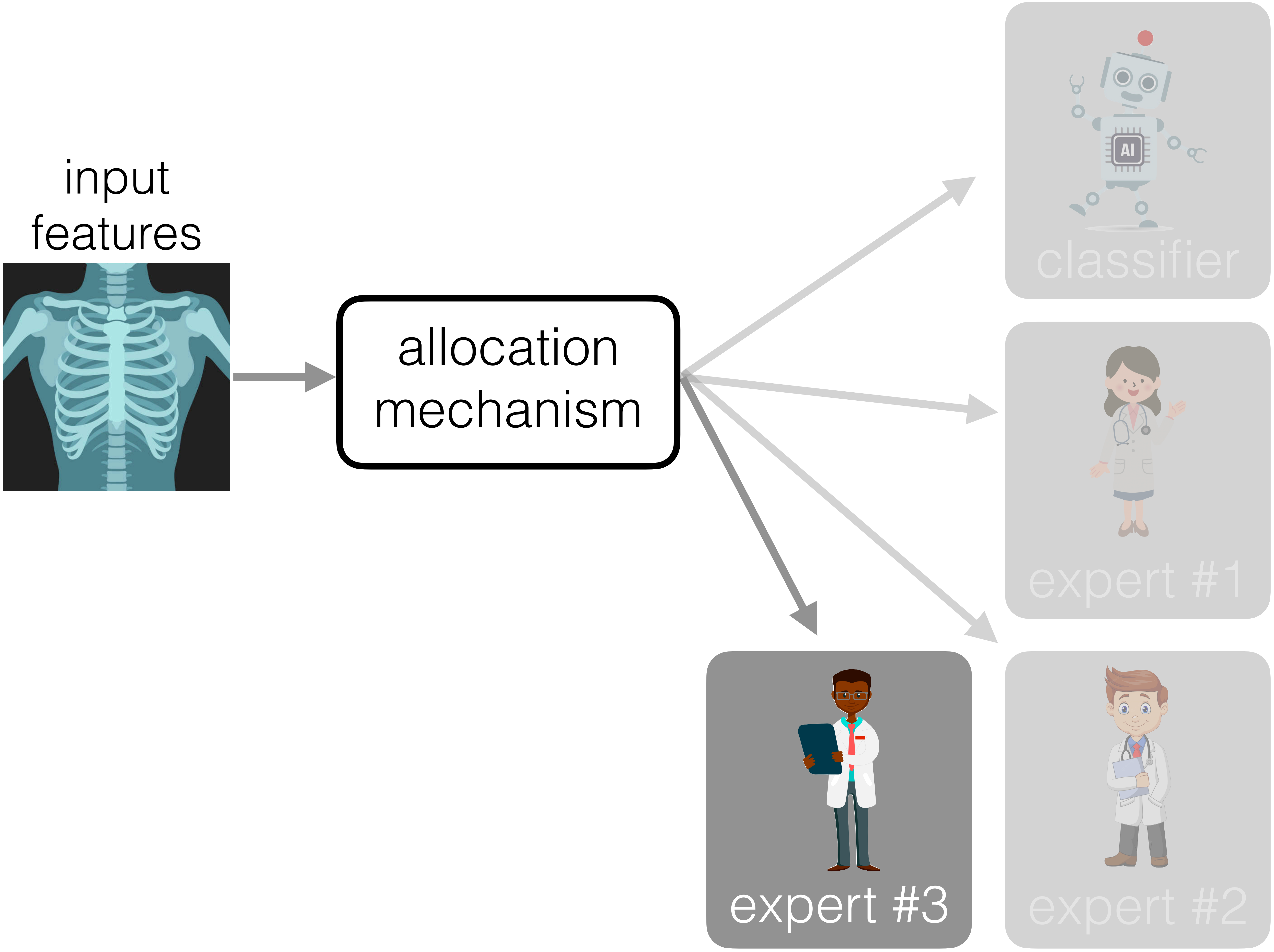
input
features
allocation
mechanism
AI
classifier
expert

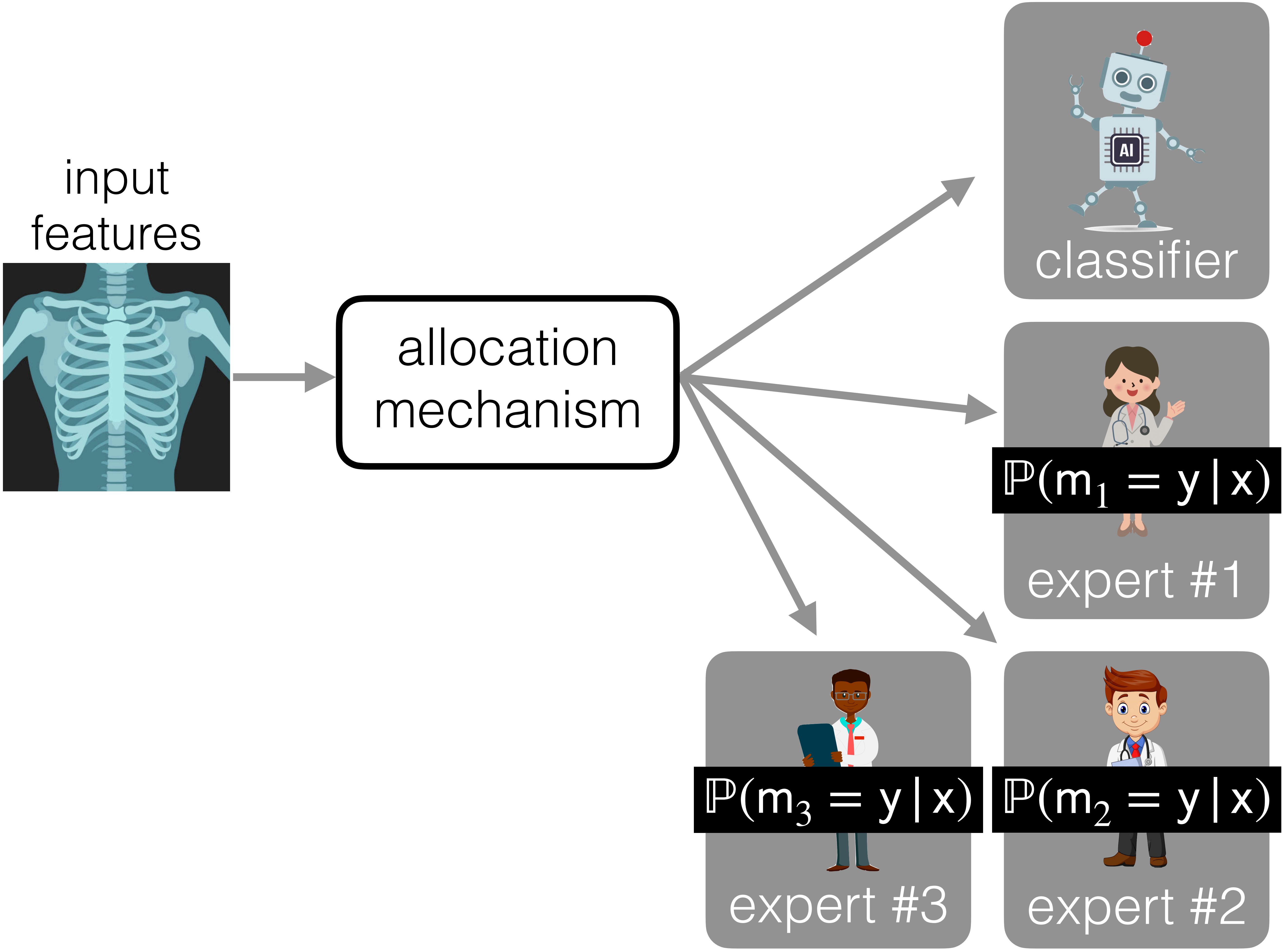

input
features
allocation
mechanism
classifier
AI
expert #1
expert #2
expert #3


input
features
allocation
mechanism
classifier
AI
expert #1
expert #3
expert #2

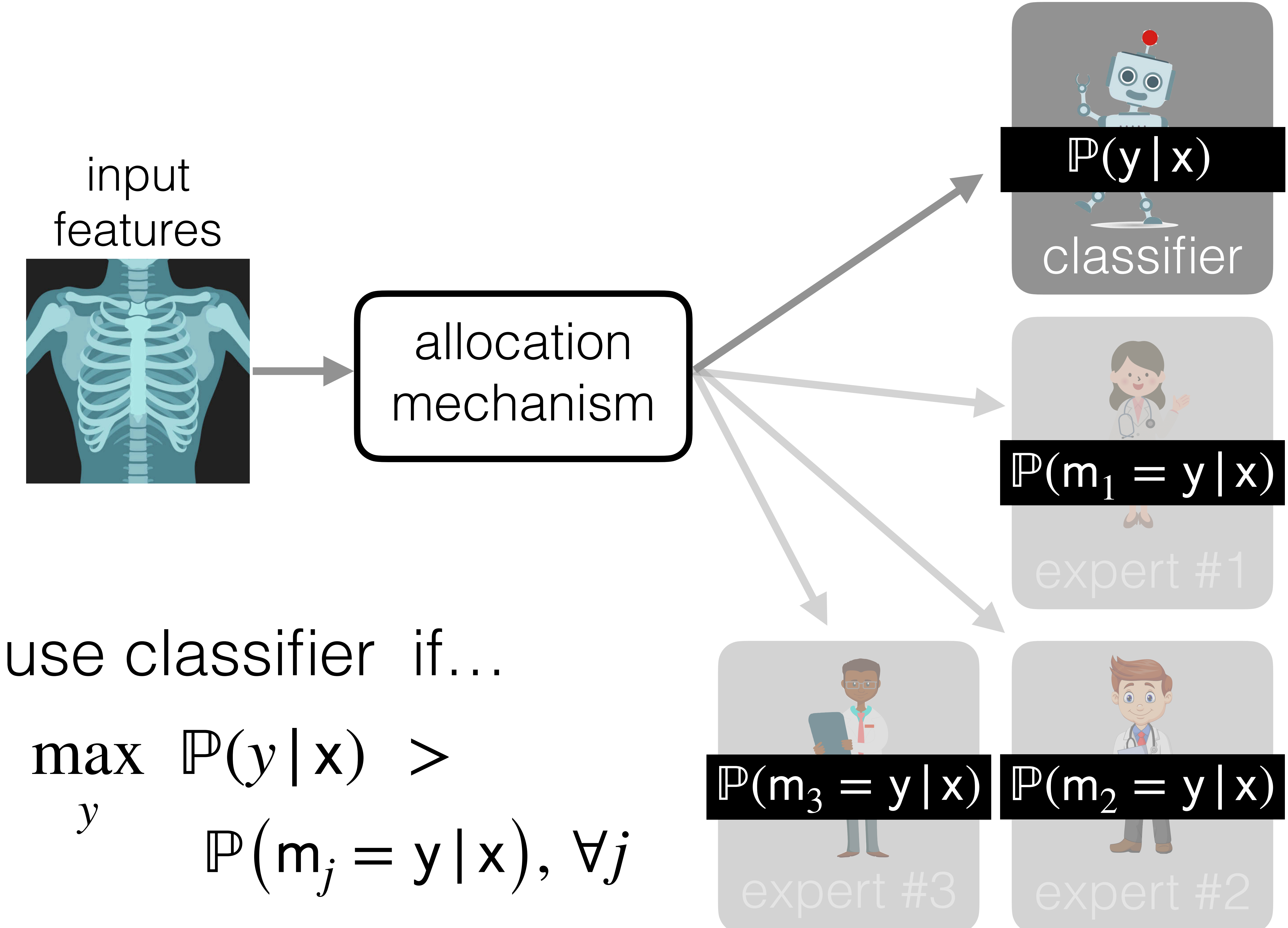

input
features
allocation
mechanism
classifier
expert #1
expert #3
expert #2


input
features
allocation
mechanism
?
classifier
AI
expert #1
expert #3
expert #2

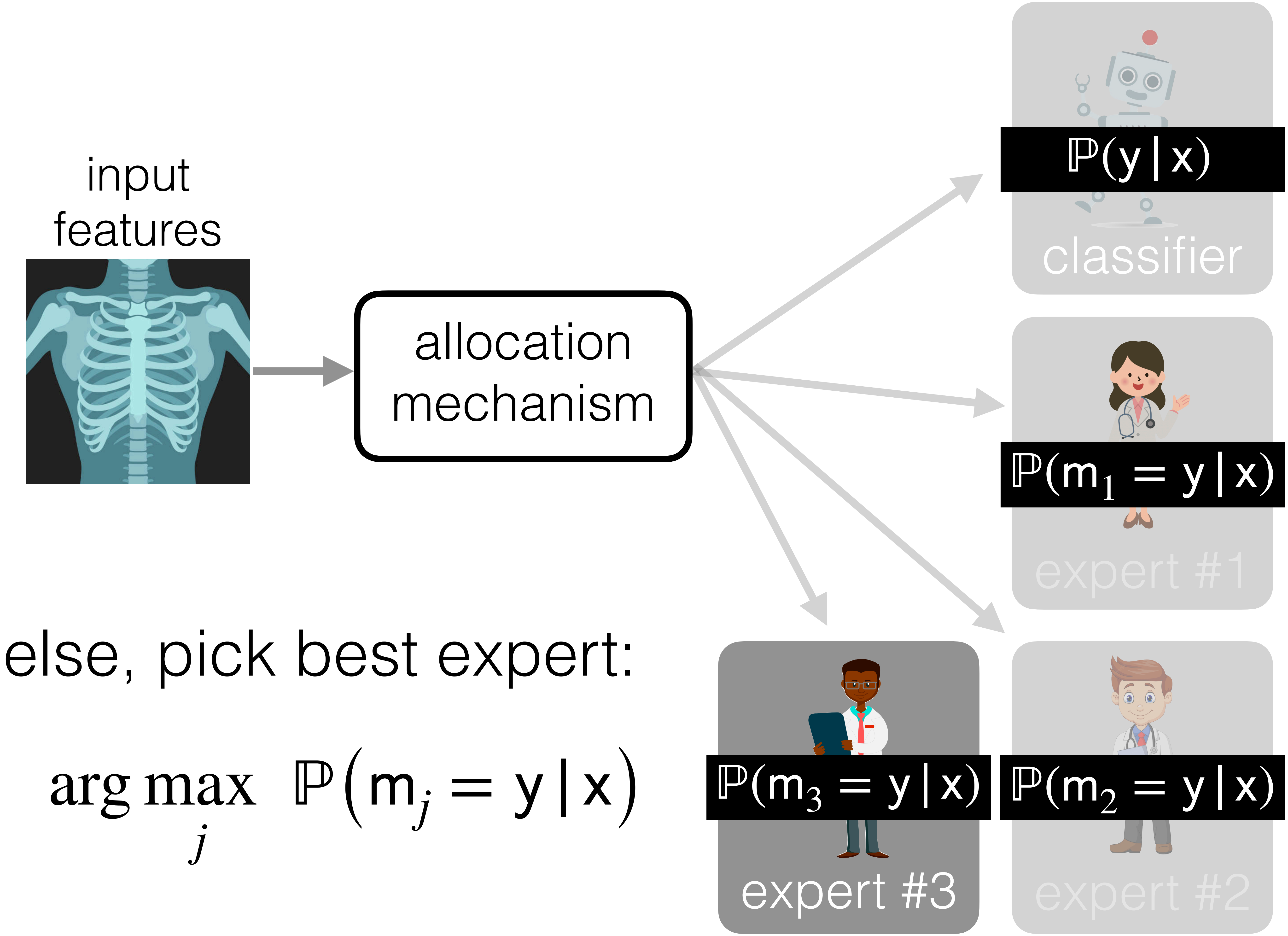

input features
allocation mechanism
classifier
expert #1
expert #2
expert #3



input
features
allocation
mechanism
classifier
$\mathbb{P}(m_1 = y \mid x)$
expert #1
$\mathbb{P}(m_3 = y \mid x)$
expert #3
$\mathbb{P}(m_2 = y \mid x)$
expert #2



input features
allocation mechanism
ℙ(y | x)
classifier
ℙ(m1 = y | x)
expert #1
ℙ(m3 = y | x)
expert #3
ℙ(m2 = y | x)
expert #2
use classifier if…
max_y ℙ(y | x) > ℙ(m_j = y | x), ∀j


input features
allocation mechanism
ℙ(y | x)
classifier
ℙ(m1 = y | x)
expert #1
ℙ(m3 = y | x)
expert #3
ℙ(m2 = y | x)
expert #2
else, pick best expert:
$\arg\max_j \mathbb{P}\left(m_j = y \mid x\right)$

multi-expert implementation

training data

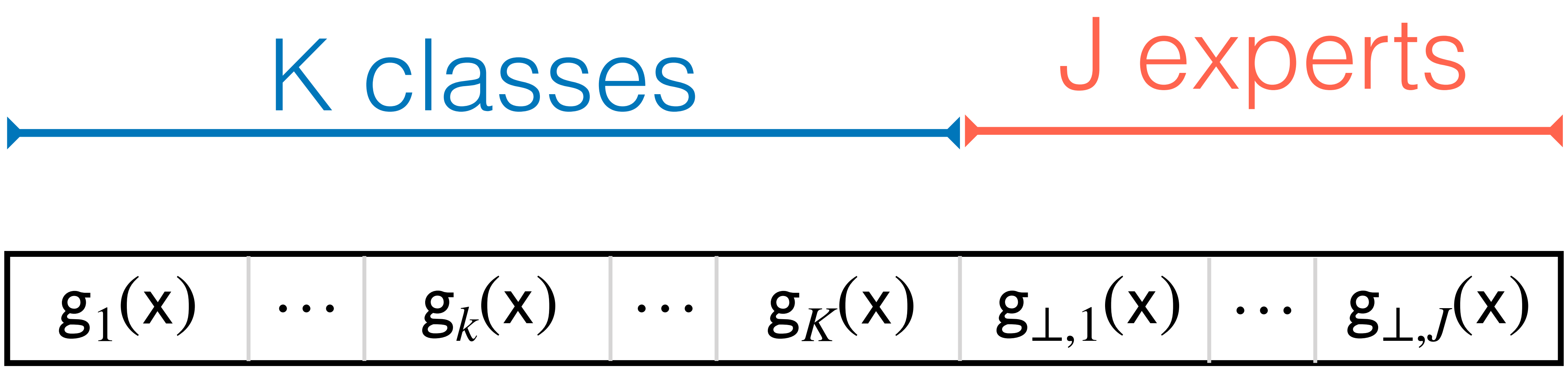
$$\mathfrak{D} = \left\{\mathsf{x}_n, \mathsf{y}_n, \mathsf{m}_{n,1}, \ldots, \mathsf{m}_{n,J}\right\}_{n=1}^{N}$$

multi-expert implementation

training data

$$\mathfrak{D} = \left\{ \mathbf{x}_n, \mathbf{y}_n, \mathbf{m}_{n,1}, \ldots, \mathbf{m}_{n,J} \right\}_{n=1}^{N}$$

model

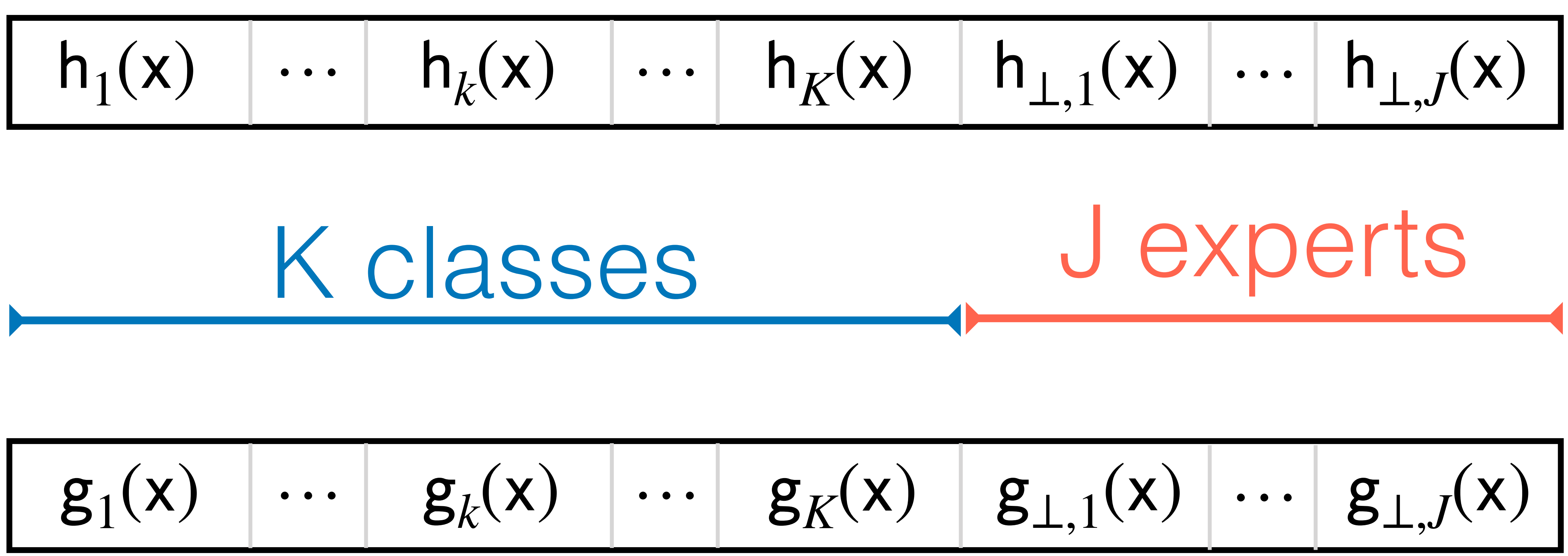


multi-expert implementation

training data

$$\mathfrak{D} = \{\mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_{n,1}, \ldots, \mathrm{m}_{n,J}\}_{n=1}^{N}$$

model

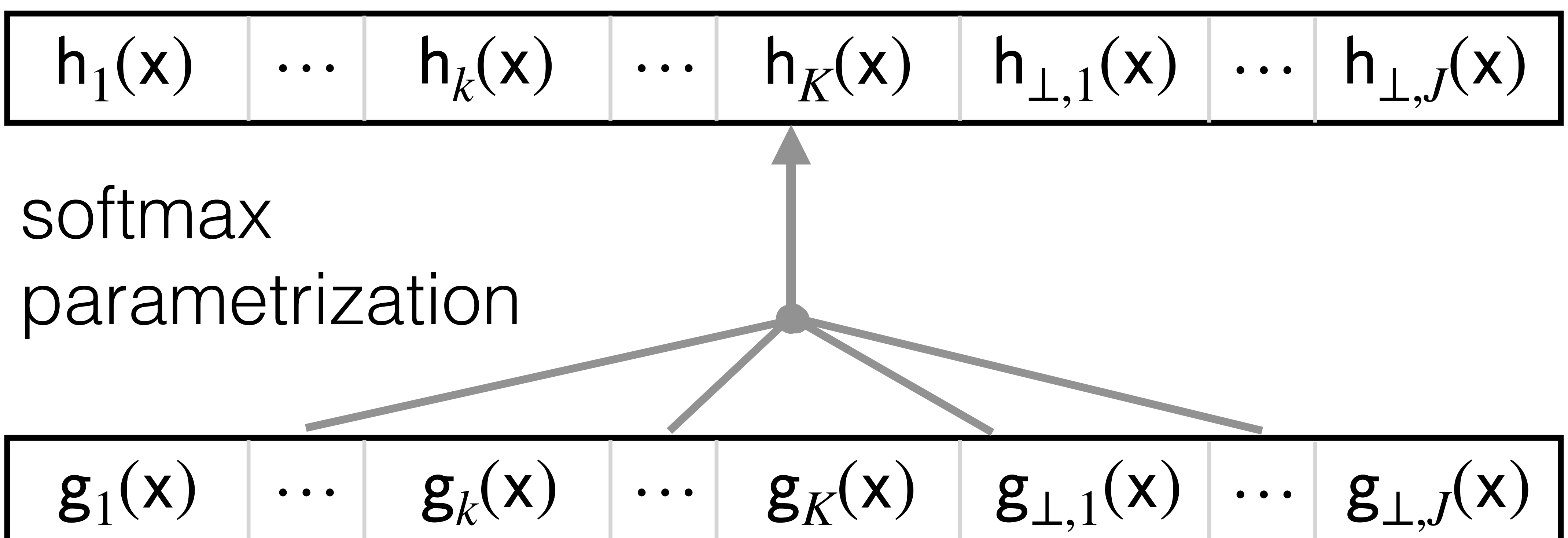


multi-expert implementation

training data

$$\mathfrak{D} = \left\{ \mathsf{x}_n, \mathsf{y}_n, \mathsf{m}_{n,1}, \ldots, \mathsf{m}_{n,J} \right\}_{n=1}^{N}$$

model

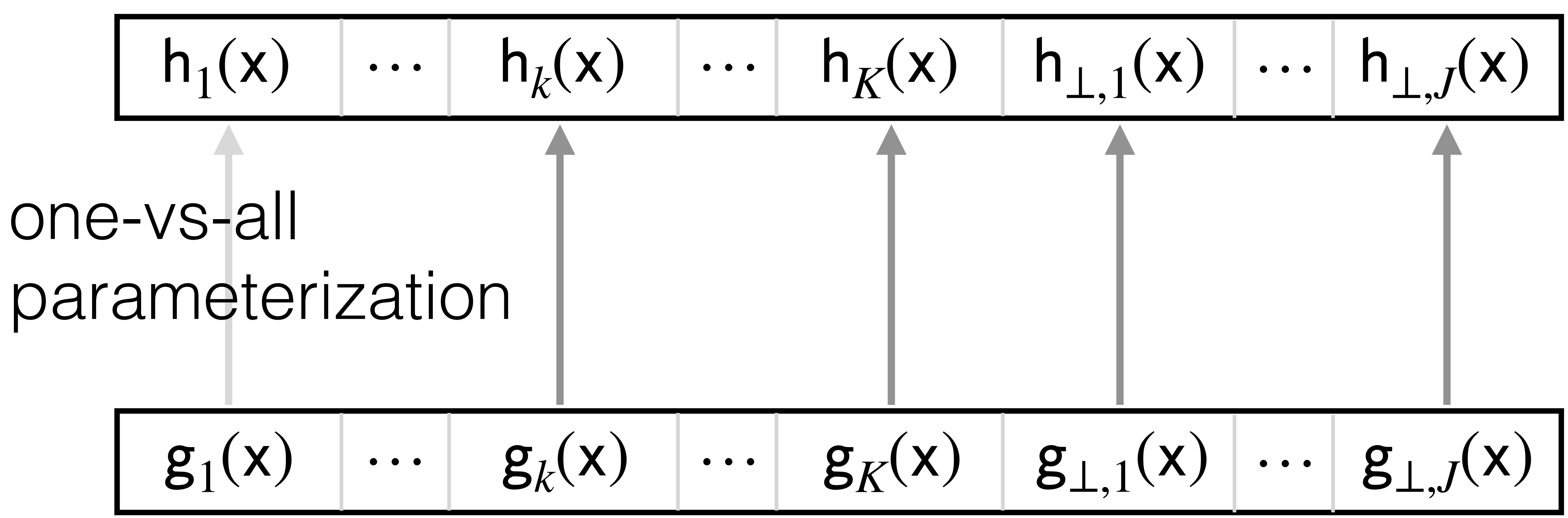


multi-expert implementation

training data

$$\mathfrak{D} = \left\{ \mathrm{x}_n, \mathrm{y}_n, \mathrm{m}_{n,1}, \ldots, \mathrm{m}_{n,J} \right\}_{n=1}^{N}$$

model



multi-expert implementation

training data

$$\mathfrak{D} = \left\{ \mathbf{x}_n, \mathbf{y}_n, \mathbf{m}_{n,1}, \ldots, \mathbf{m}_{n,J} \right\}_{n=1}^{N}$$

model

- ⊗ softmax and one-vs-all variants
- ⊗ both consistent w.r.t. 0-1 loss

multi-expert implementation

training data

$$\mathfrak{D} = \{\mathsf{x}_n, \mathsf{y}_n, \mathsf{m}_{n,1}, \ldots, \mathsf{m}_{n,J}\}_{n=1}^{N}$$

model

- ⊗ softmax and one-vs-all variants
- ⊗ both consistent w.r.t. 0-1 loss

softmax loss function

$$\ell(\theta; \mathsf{x}, \mathsf{y}, \mathsf{m}) = -\log \mathsf{h}_{\mathsf{y}}(\mathsf{x}) - \sum_j \mathbb{I}\left[\mathsf{y} = \mathsf{m}_j\right] \cdot \log \mathsf{h}_{\perp,j}(\mathsf{x})$$

multi-expert implementation

training data

$$\mathfrak{D} = \left\{ \mathsf{x}_n, \mathsf{y}_n, \mathsf{m}_{n,1}, \ldots, \mathsf{m}_{n,J} \right\}_{n=1}^{N}$$

model

- softmax and one-vs-all variants
- both consistent w.r.t. 0-1 loss

softmax loss function

$$\ell(\theta; \mathsf{x}, \mathsf{y}, \mathsf{m}) = -\log \mathsf{h}_{\mathsf{y}}(\mathsf{x}) - \sum_{j} \mathbb{I}\left[\mathsf{y} = \mathsf{m}_j\right] \cdot \log \mathsf{h}_{\perp,j}(\mathsf{x})$$

multi-expert implementation

training data

$$\mathfrak{D} = \left\{ \mathsf{x}_n, \mathsf{y}_n, \mathsf{m}_{n,1}, \ldots, \mathsf{m}_{n,J} \right\}_{n=1}^{N}$$

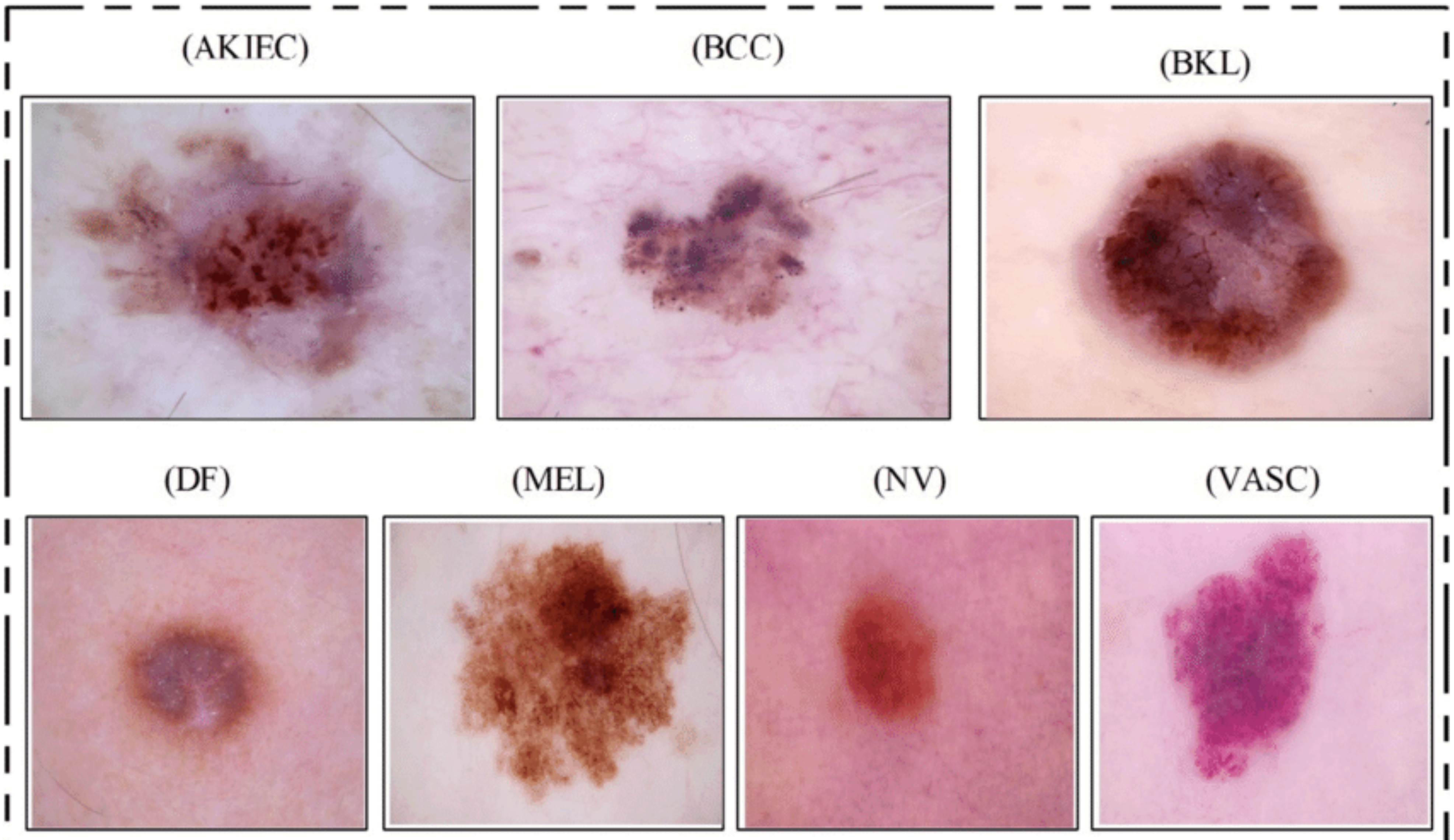
model

- ⊗ softmax and one-vs-all variants
- ⊗ both consistent w.r.t. 0-1 loss

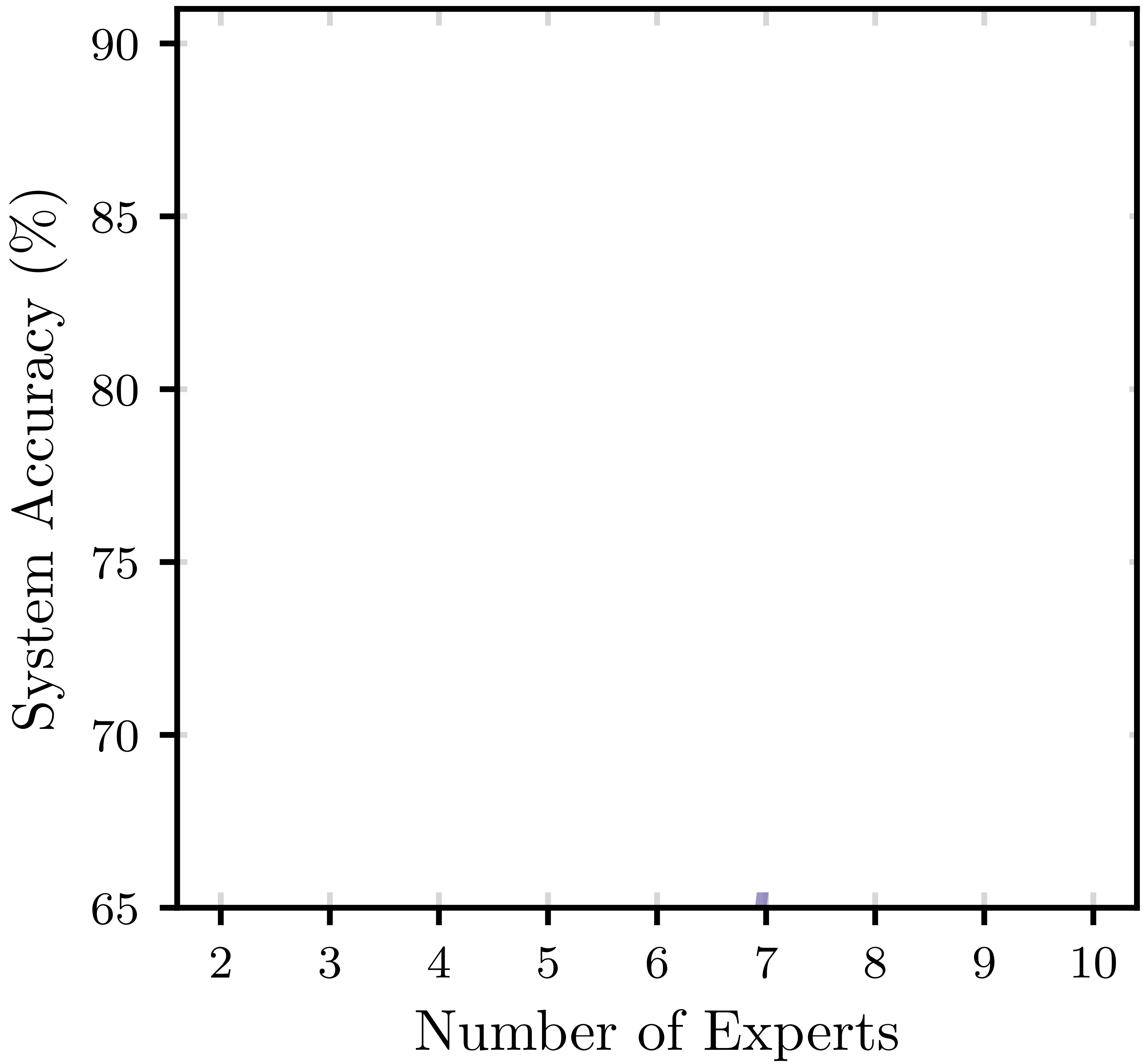
softmax loss function

$$\ell(\theta; \mathsf{x}, \mathsf{y}, \mathsf{m}) = -\log \mathsf{h}_{\mathsf{y}}(\mathsf{x}) - \sum_j \mathbb{I}\left[\mathsf{y} = \mathsf{m}_j\right] \cdot \log \mathsf{h}_{\perp,j}(\mathsf{x})$$

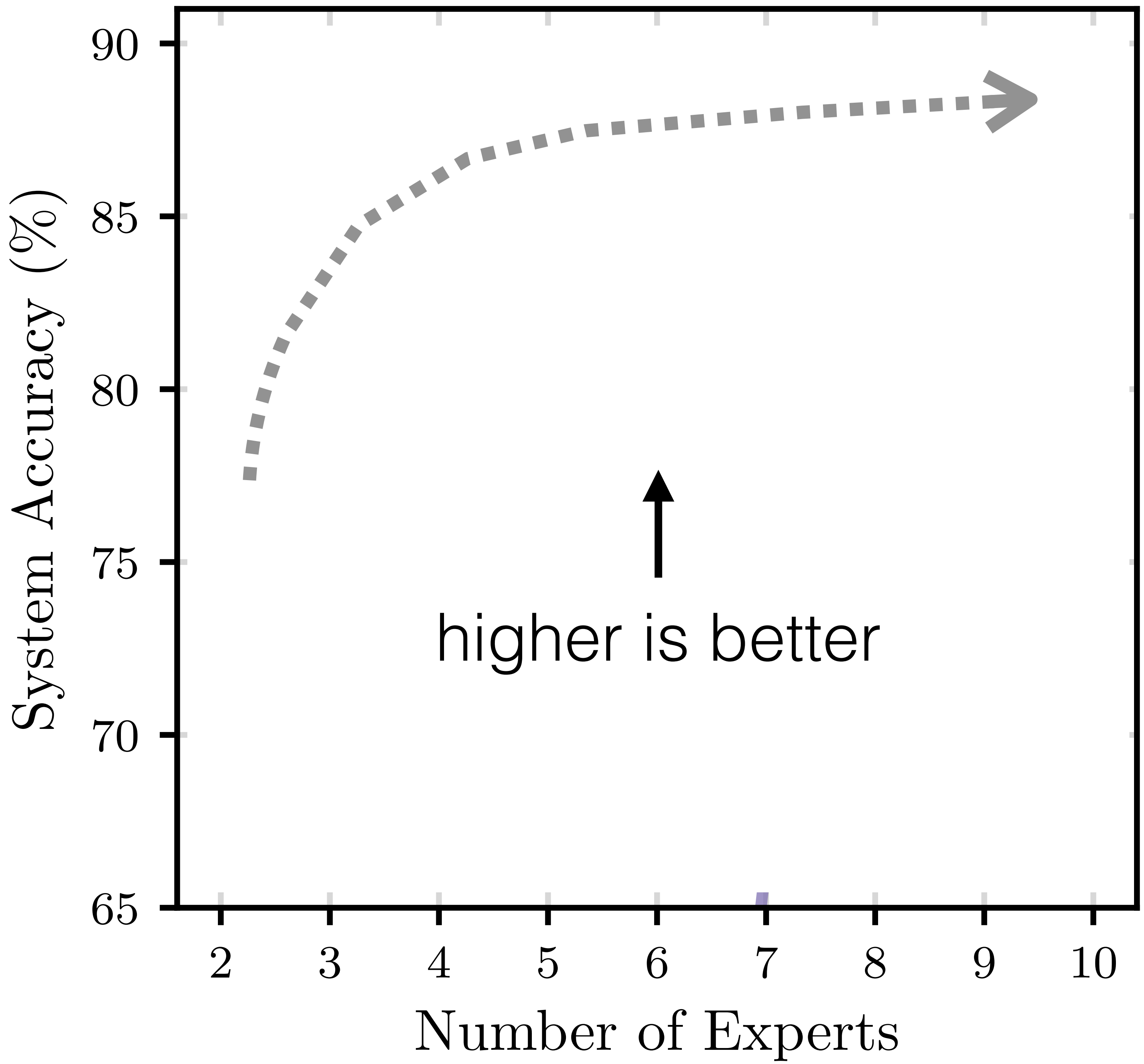
skin lesion diagnosis



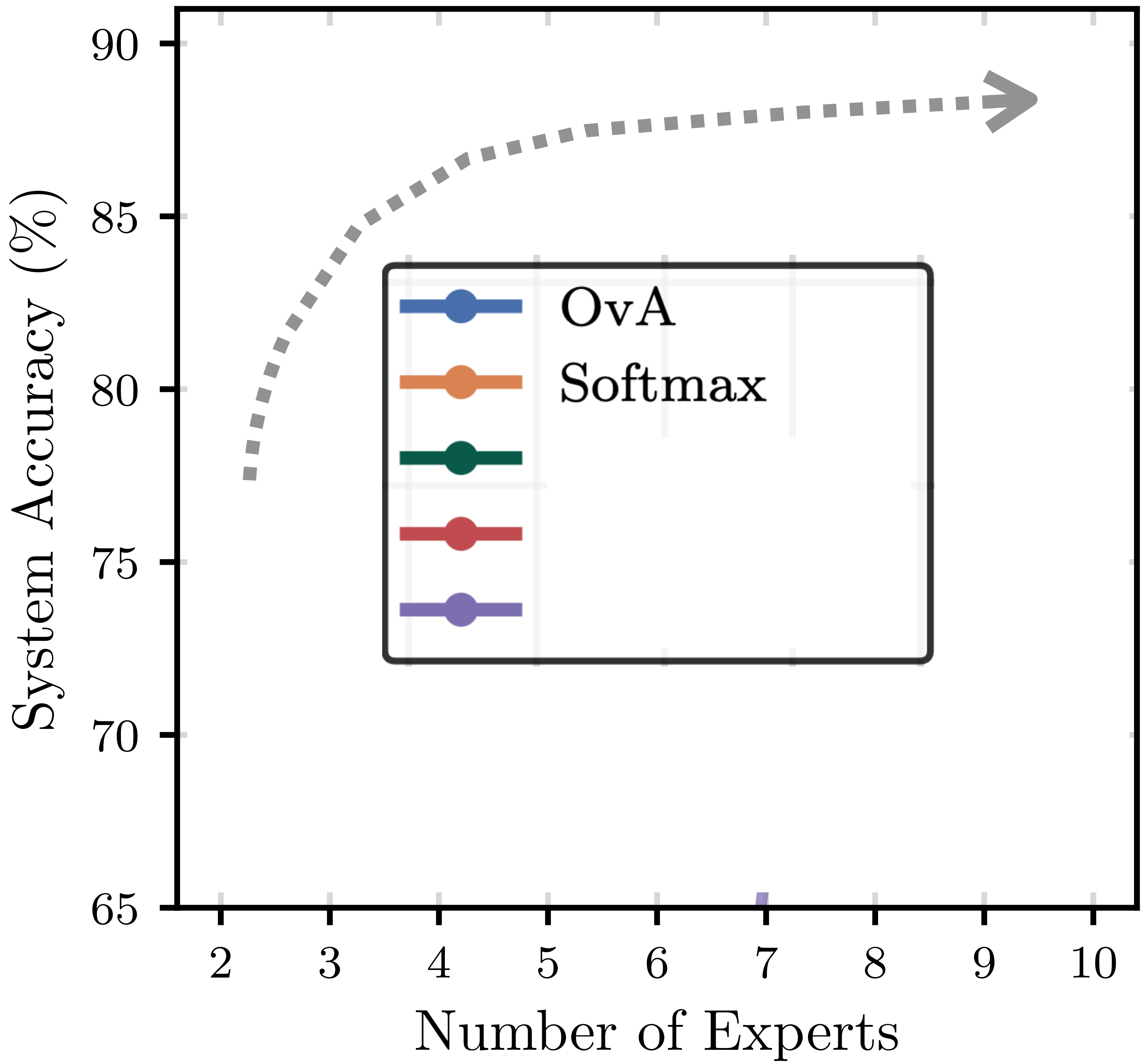
skin lesion diagnosis



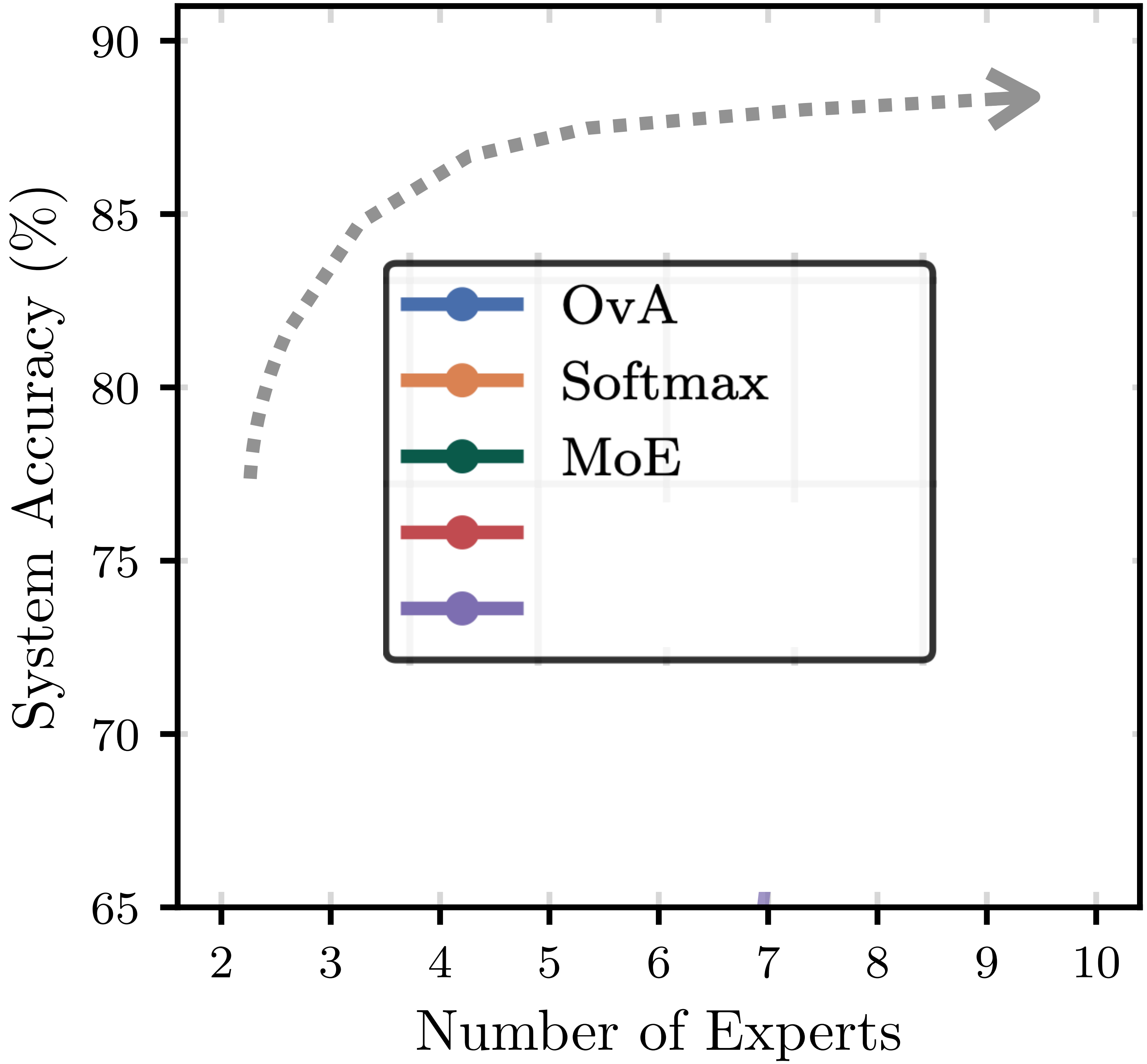
skin lesion diagnosis



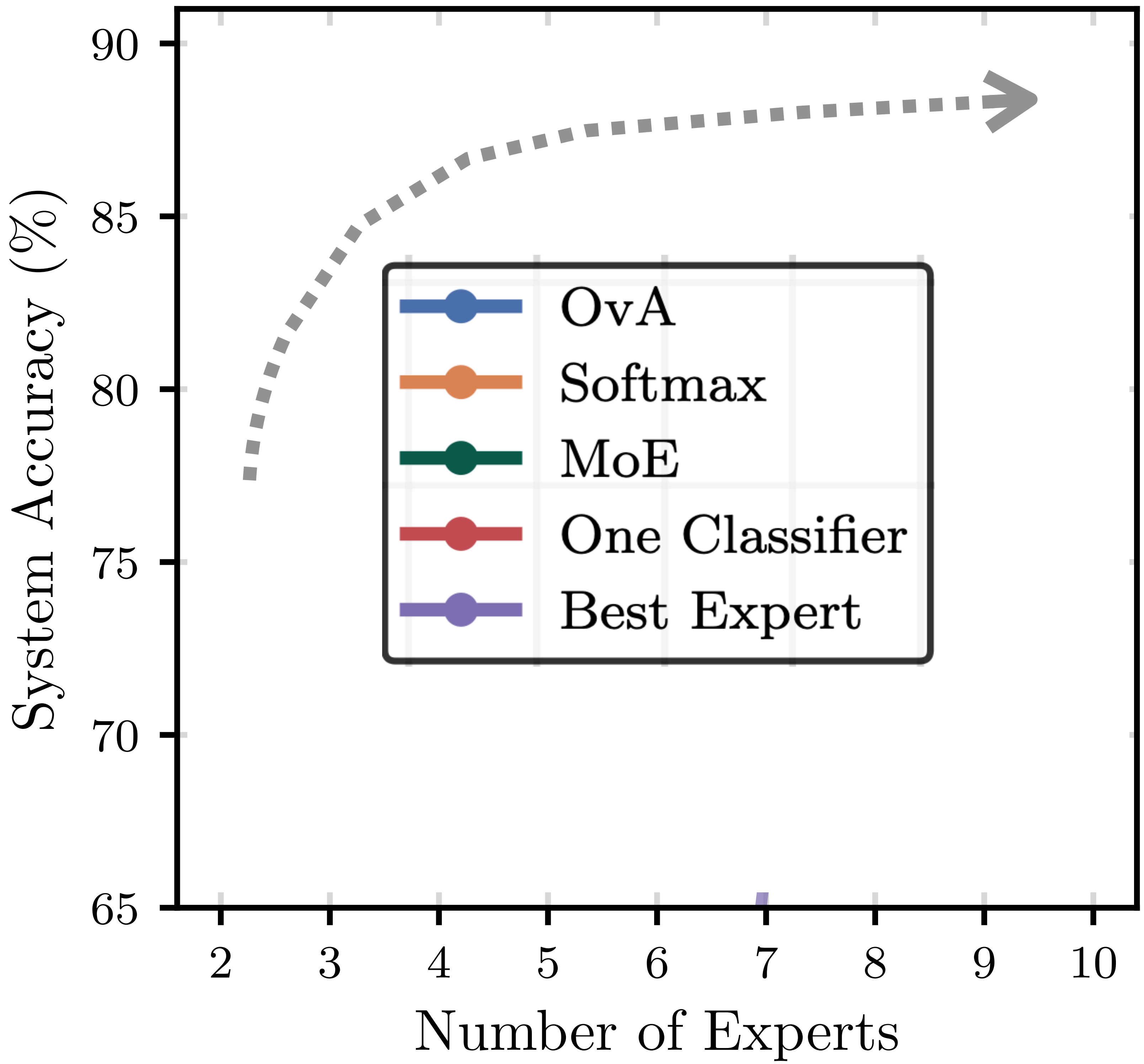
skin lesion diagnosis

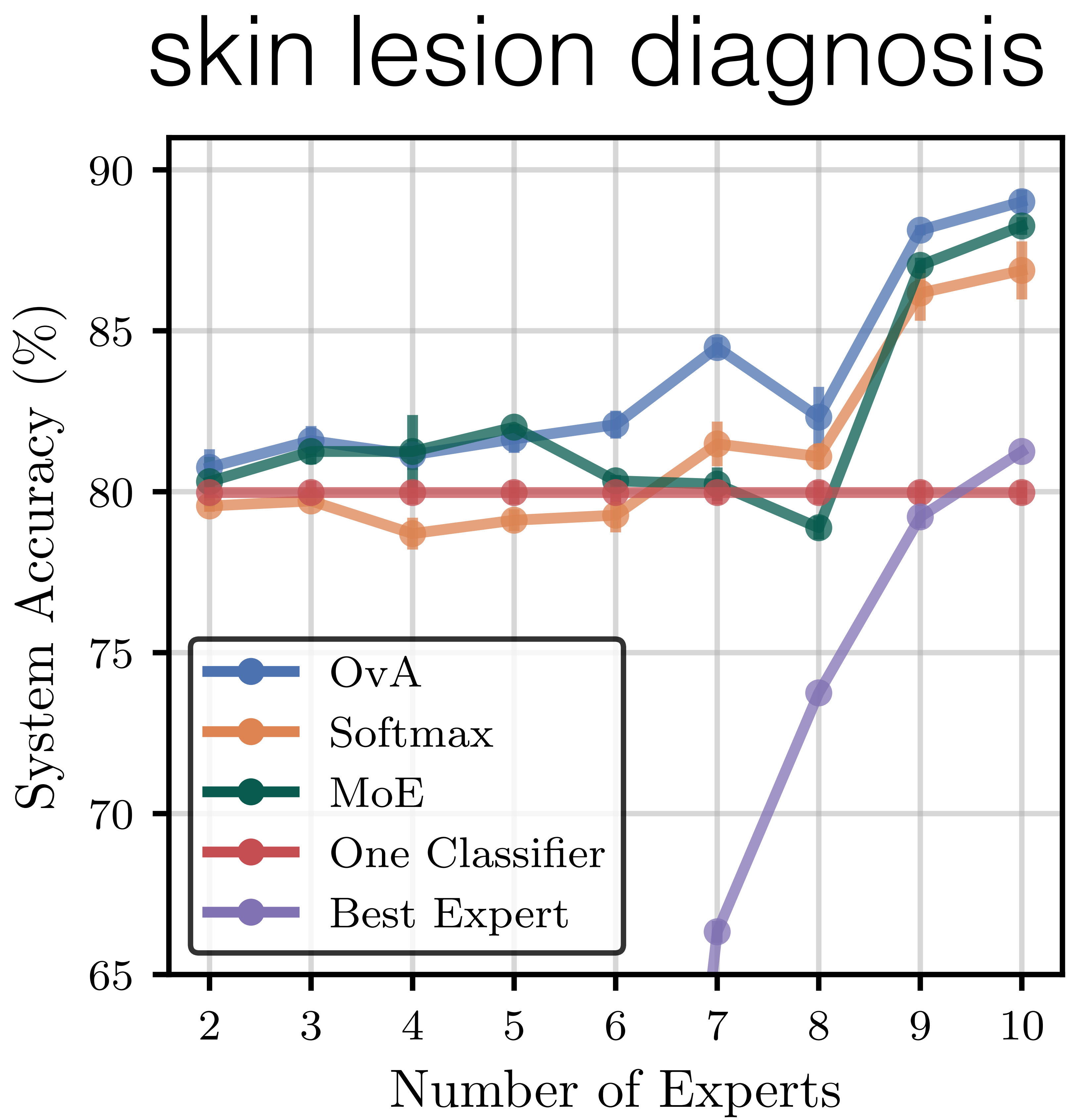


skin lesion diagnosis

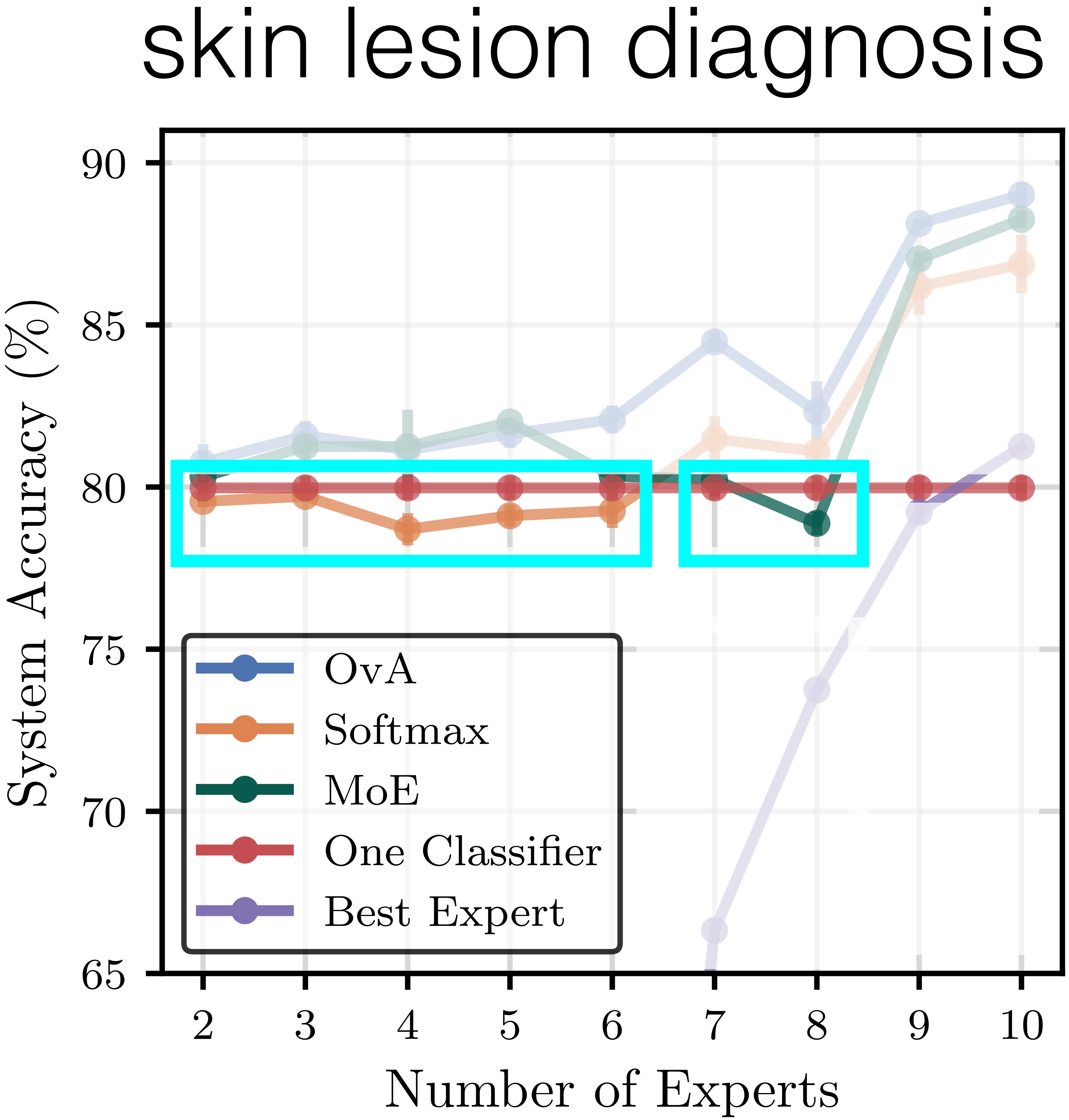


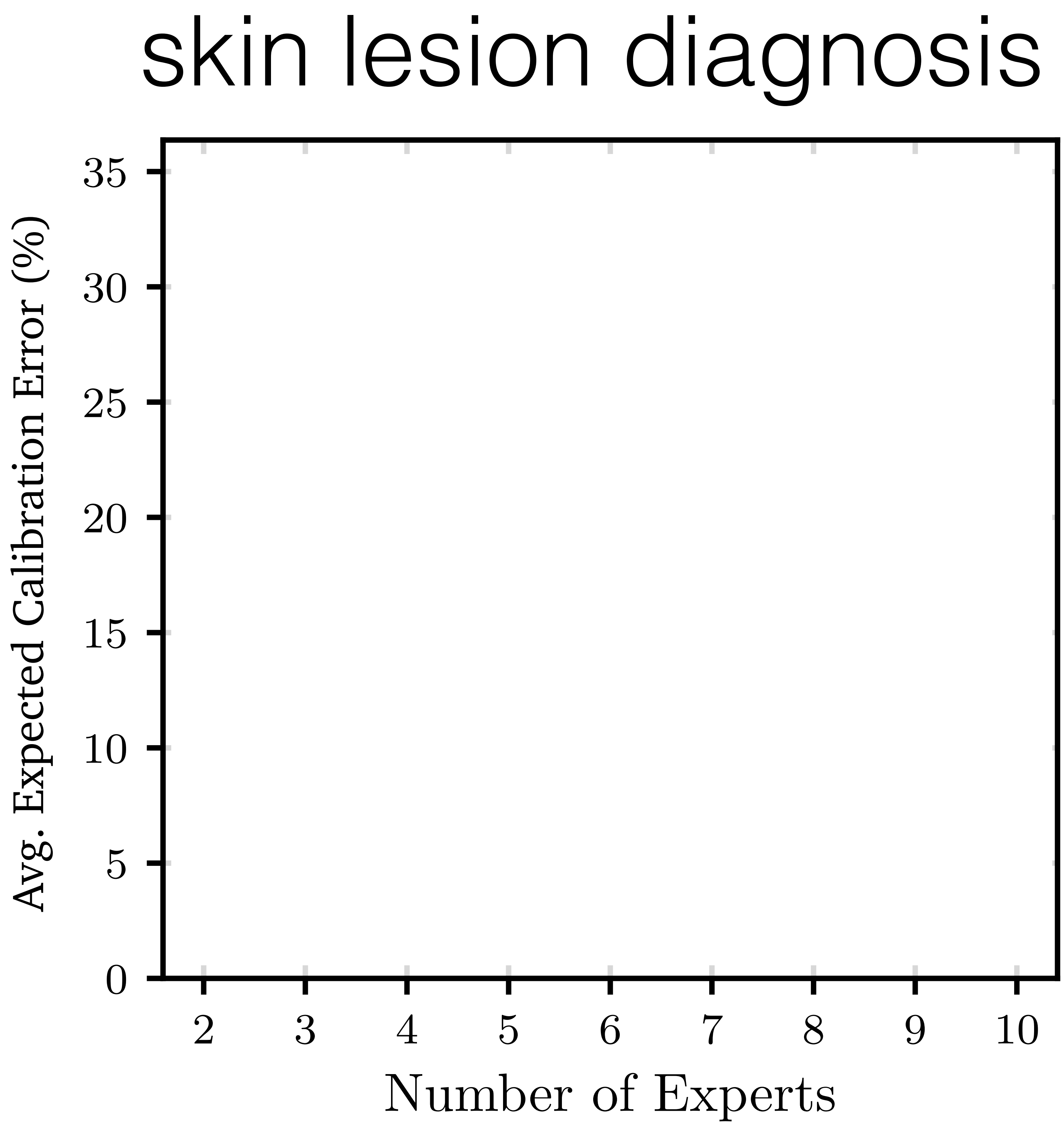
skin lesion diagnosis



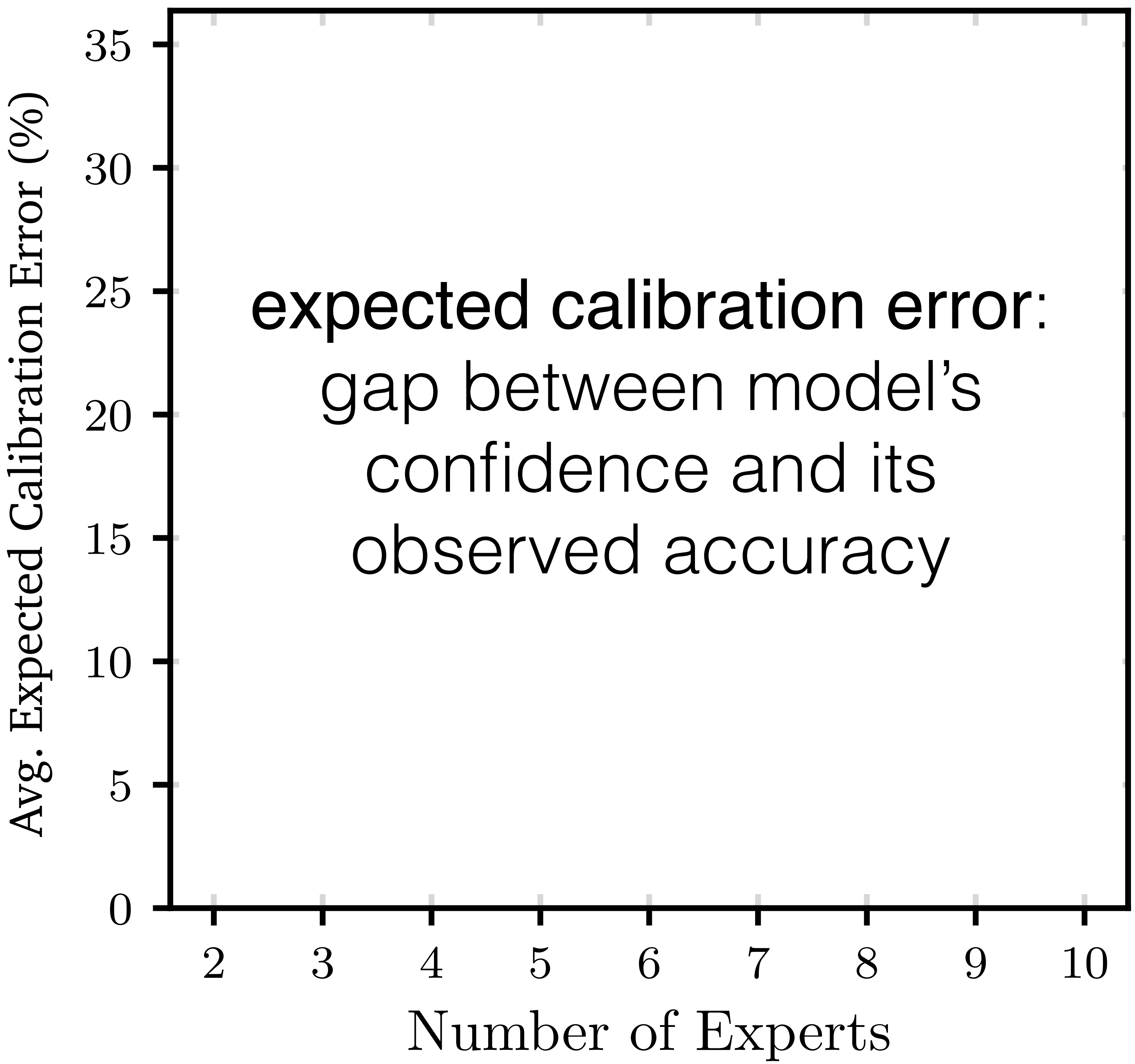


skin lesion diagnosis
System Accuracy (%)
Number of Experts
90
85
80
75
70
65
2
3
4
5
6
7
8
9
10
OvA
Softmax
MoE
One Classifier
Best Expert

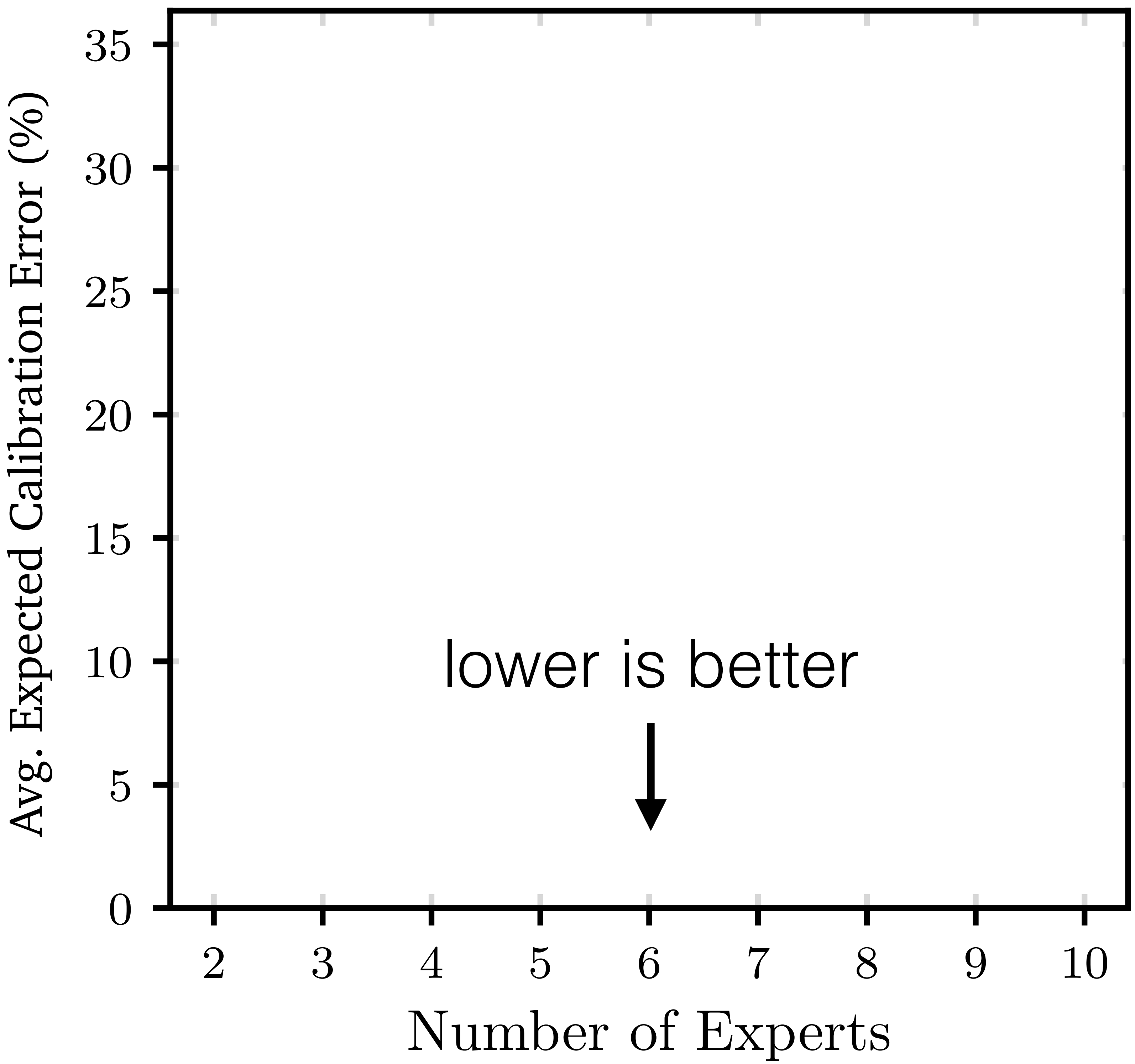

skin lesion diagnosis
System Accuracy (%)
90
85
80
75
70
65
2
3
4
5
6
7
8
9
10
Number of Experts
OvA
Softmax
MoE
One Classifier
Best Expert

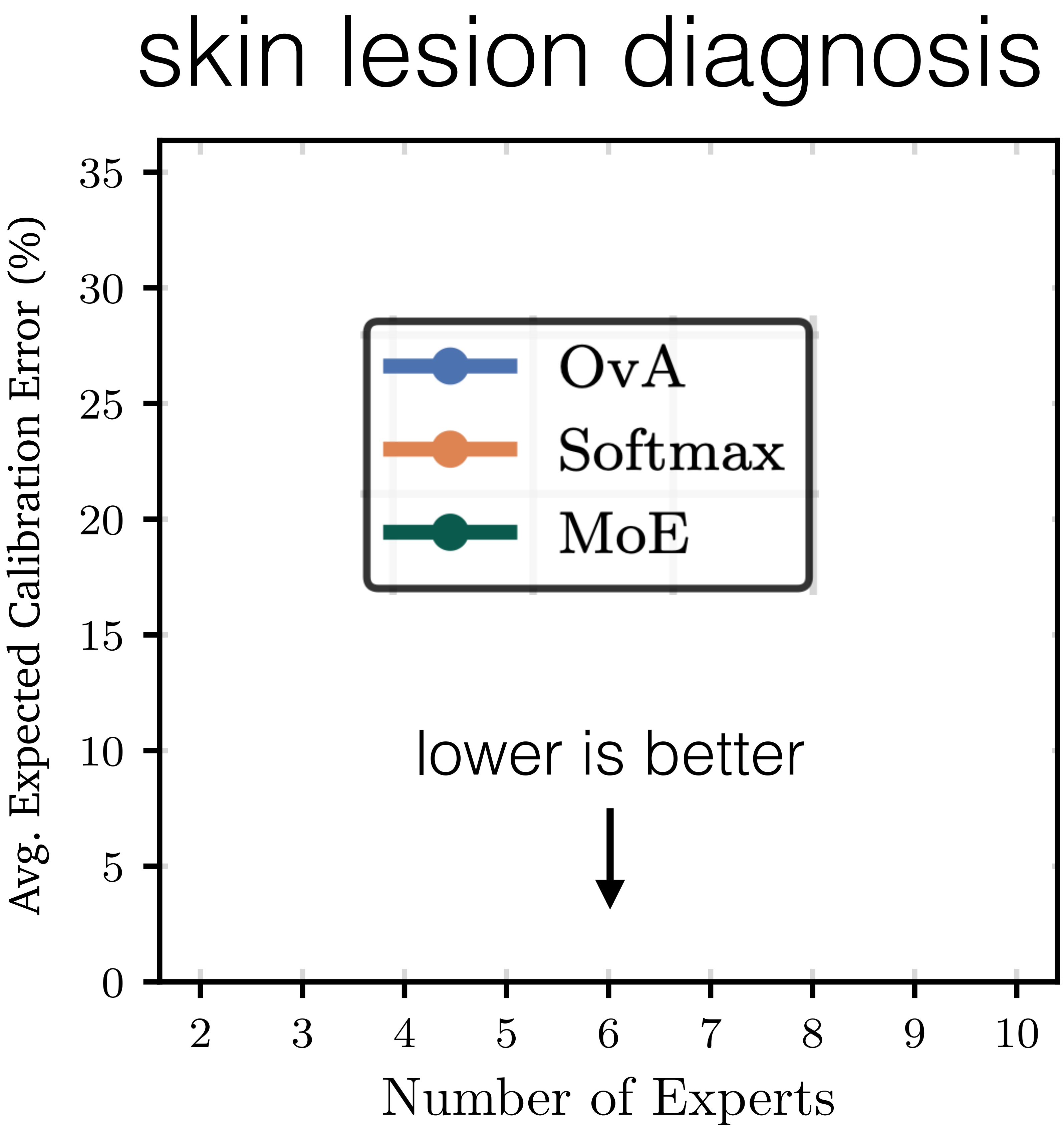

skin lesion diagnosis
Avg. Expected Calibration Error (%)
0
5
10
15
20
25
30
35
2
3
4
5
6
7
8
9
10
Number of Experts

skin lesion diagnosis

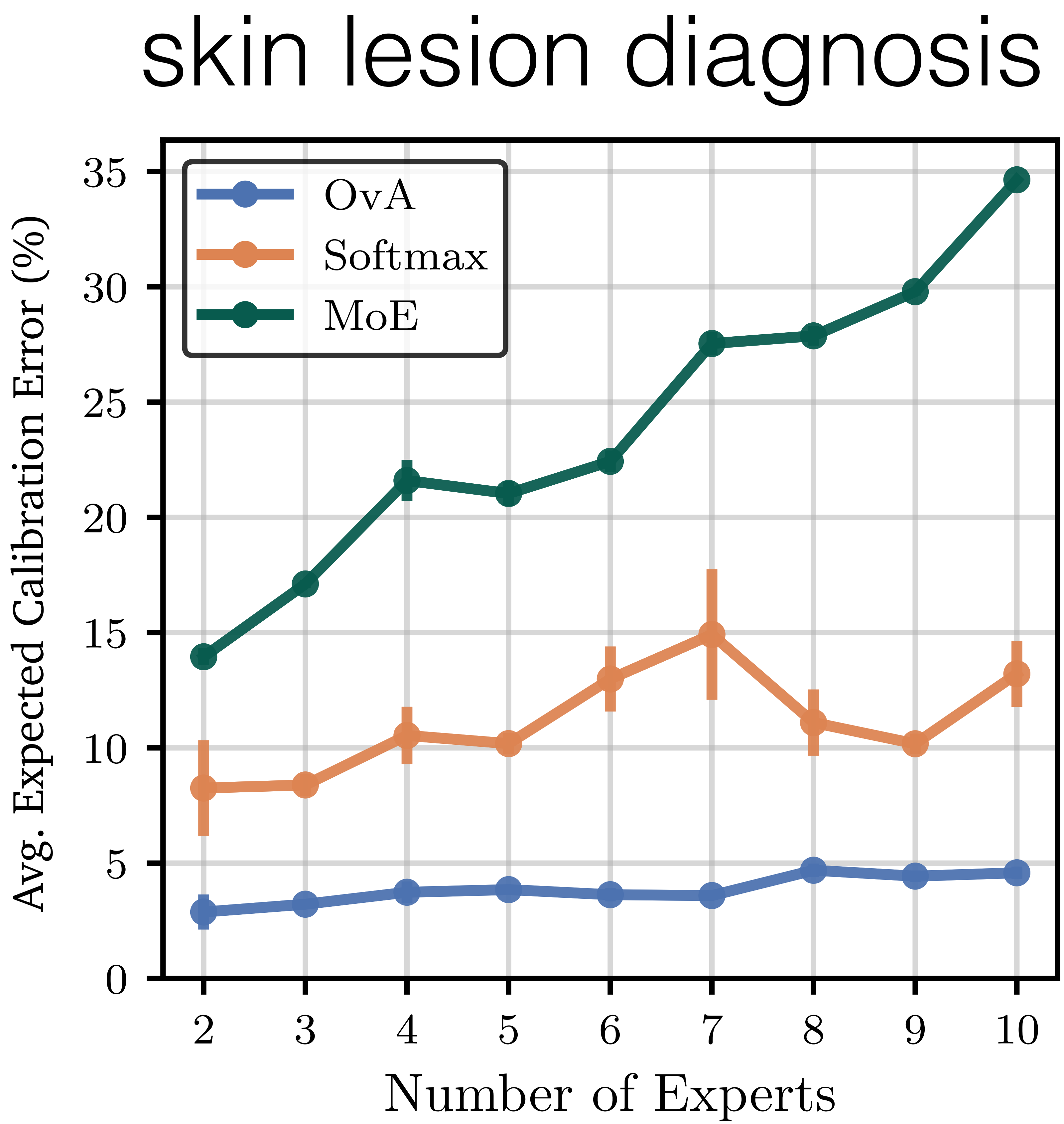


skin lesion diagnosis





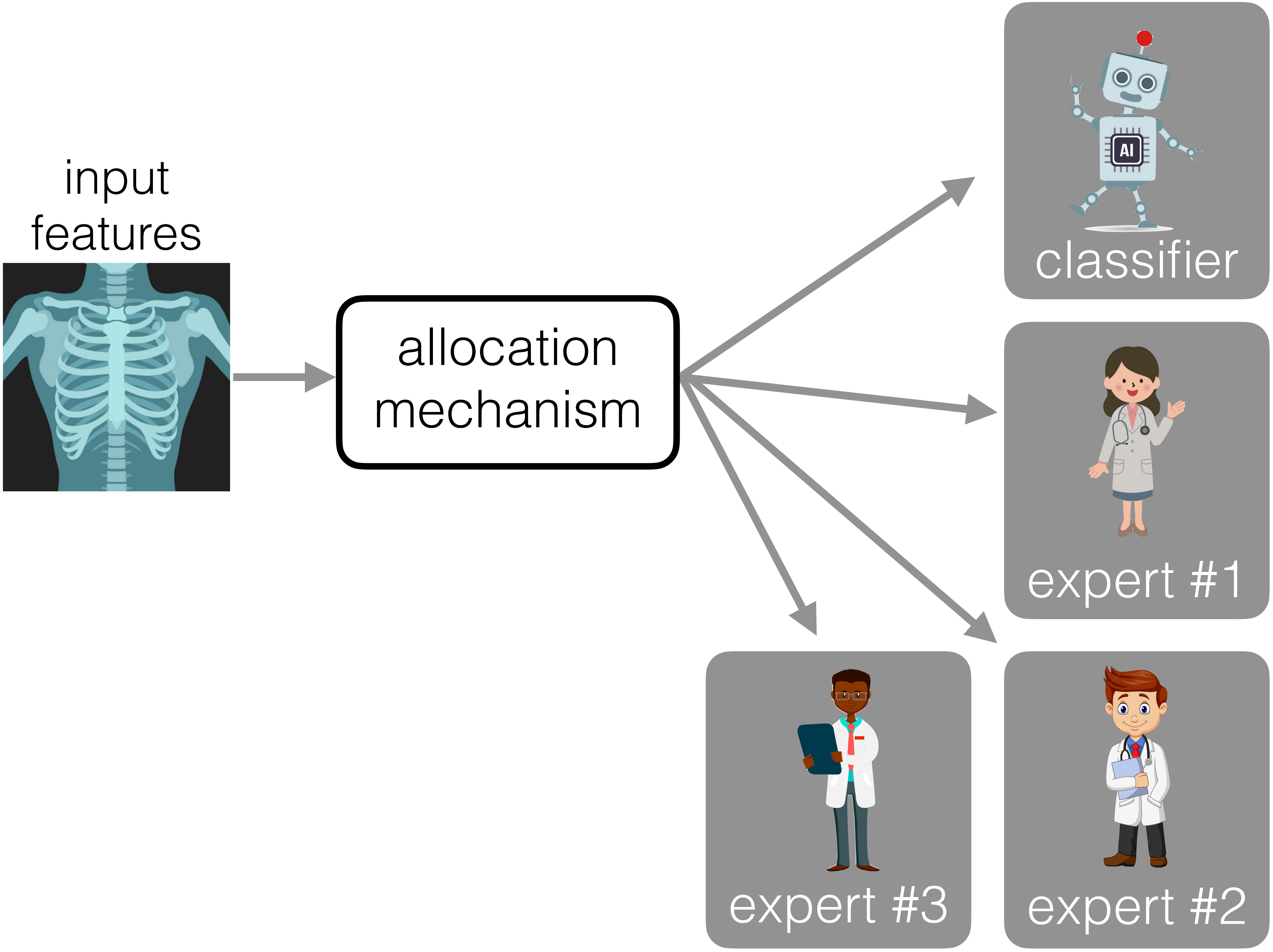
skin lesion diagnosis
Avg. Expected Calibration Error (%)
35
30
25
20
15
10
5
0
OvA
Softmax
MoE
lower is better
2
3
4
5
6
7
8
9
10
Number of Experts

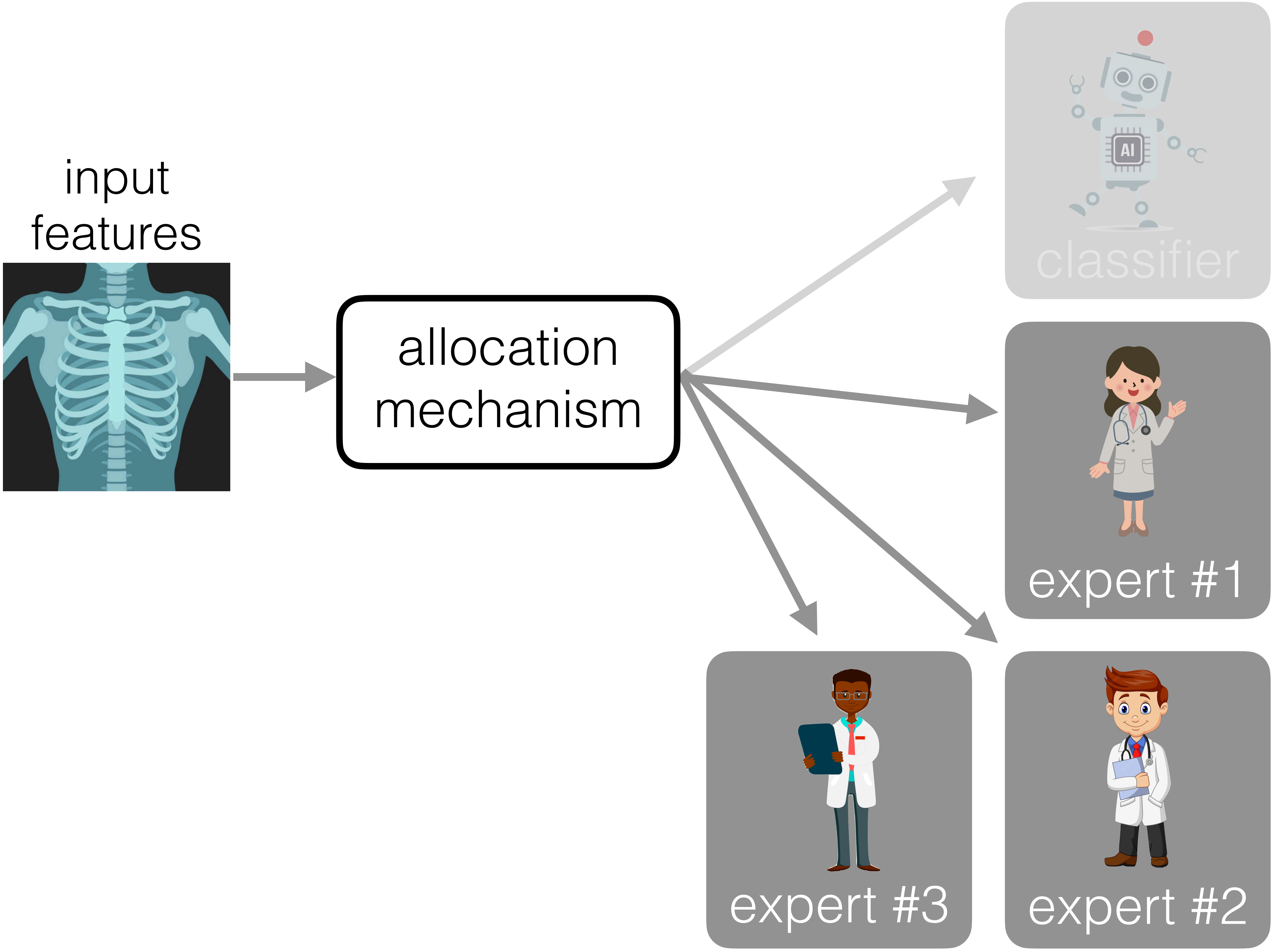


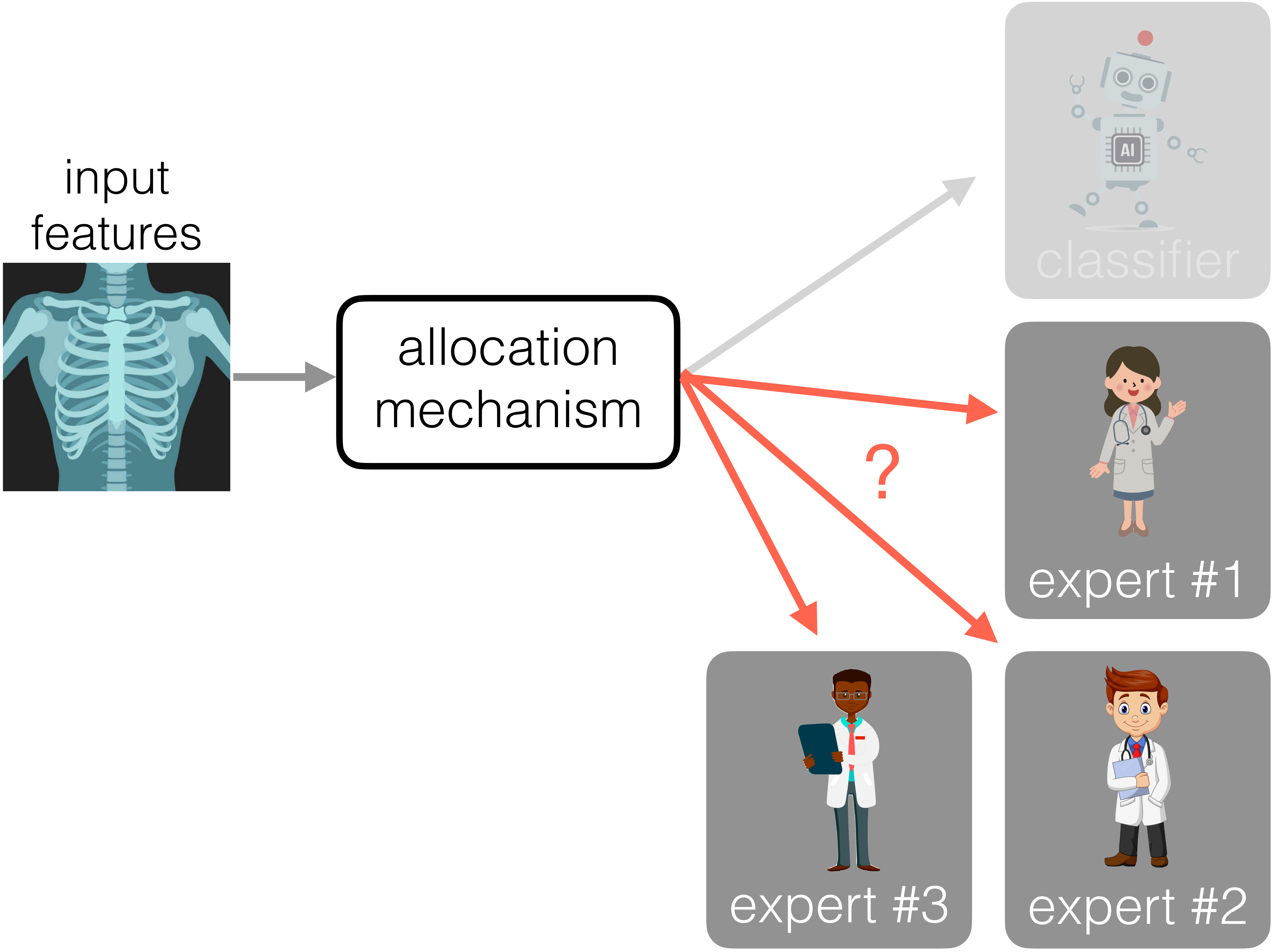
skin lesion diagnosis
OvA
Softmax
MoE
Avg. Expected Calibration Error (%)
0
5
10
15
20
25
30
35
2
3
4
5
6
7
8
9
10
Number of Experts

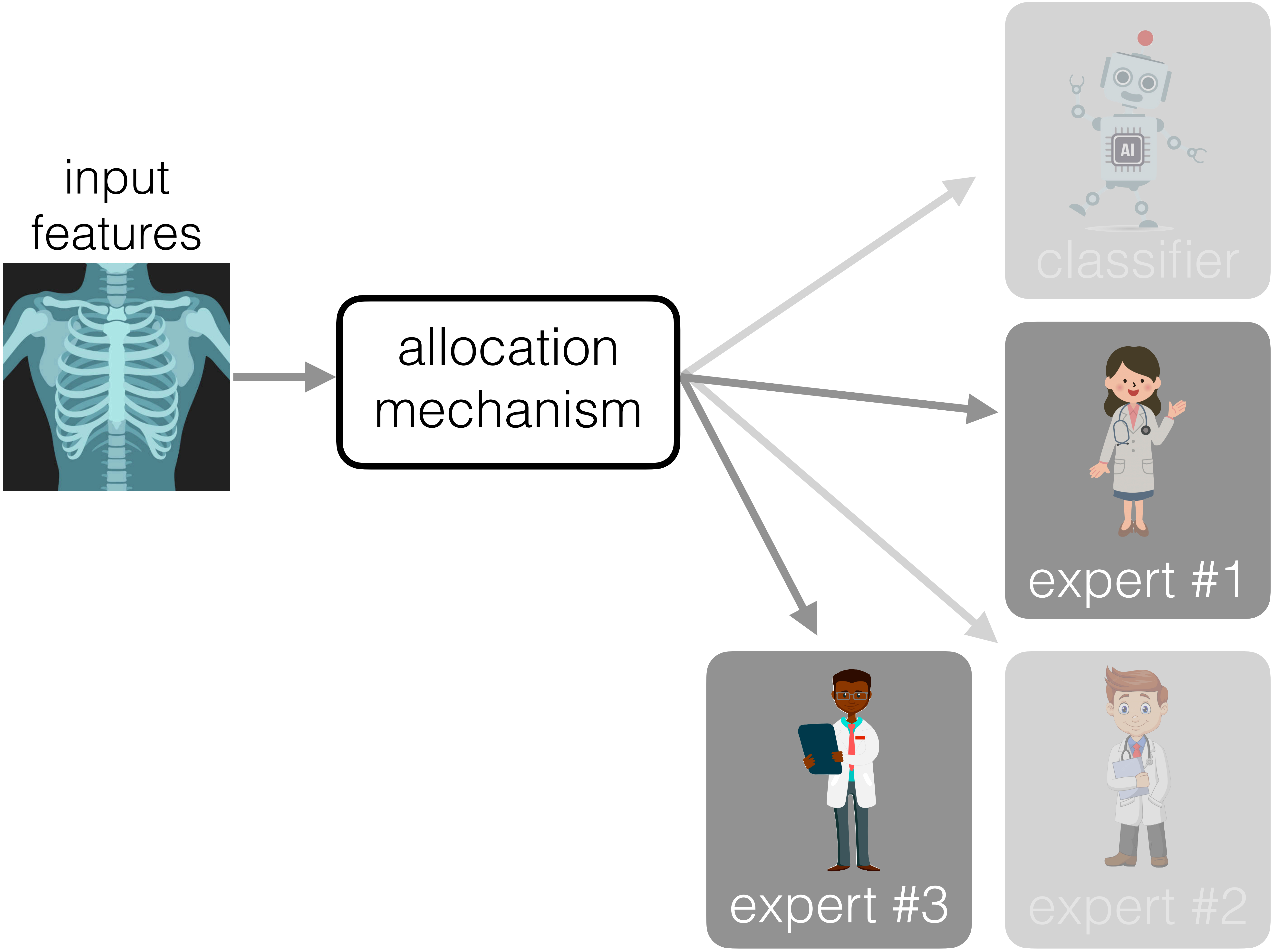
- single expert
  - softmax surrogate loss
  - improving calibration via one-vs-all
- **multiple experts**
  - surrogate losses
  - conformal sets of experts
- population of experts
  - surrogate losses
  - meta-learning a rejector

- single expert
  - softmax surrogate loss
  - improving calibration via one-vs-all
- **multiple experts**
  - surrogate losses
  - conformal sets of experts
- population of experts
  - surrogate losses
  - meta-learning a rejector


input
features
allocation
mechanism
classifier
AI
expert #1
expert #3
expert #2


input
features
allocation
mechanism
classifier
expert #1
expert #3
expert #2


input
features
allocation
mechanism
classifier
AI
?
expert #1
expert #3
expert #2


input
features
allocation
mechanism
classifier
AI
expert #1
expert #3
expert #2

conformal inference

conformal inference

assume there's a best expert, **j***:

$$\mathbb{P}\left(\mathsf{m}_{j*} = \mathsf{y} \,|\, \mathsf{x}\right) \;>\; \mathbb{P}\left(\mathsf{m}_{e} = \mathsf{y} \,|\, \mathsf{x}\right), \;\; \forall e \neq j*$$

conformal inference

assume there's a best expert, j*:

$$\mathbb{P}\left(\mathsf{m}_{j*} = \mathsf{y} \,|\, \mathsf{x}\right) \;>\; \mathbb{P}\left(\mathsf{m}_e = \mathsf{y} \,|\, \mathsf{x}\right), \;\; \forall e \neq j*$$

construct a confidence set of experts:

$$\mathbb{P}\left(\mathsf{j}* \in \mathsf{C}(\mathsf{x})\right) \;\geq\; 1 - \alpha$$

conformal inference

assume there's a best expert, j*:

$$\mathbb{P}\left(\mathsf{m}_{j*} = \mathsf{y} \,|\, \mathsf{x}\right) \;>\; \mathbb{P}\left(\mathsf{m}_{e} = \mathsf{y} \,|\, \mathsf{x}\right), \;\; \forall e \neq j*$$

construct a confidence set of experts:

$$\mathbb{P}\left(\mathsf{j}^* \in \mathsf{C}(\mathsf{x})\right) \;\geq\; 1 - \alpha$$

team of experts: adaptive in size and membership

conformal inference: ensembling

C(x) = { ,  }

conformal inference: ensembling

$C(x) = \{$  ,  $\}$

predictions: m_3 m_1

conformal inference: ensembling

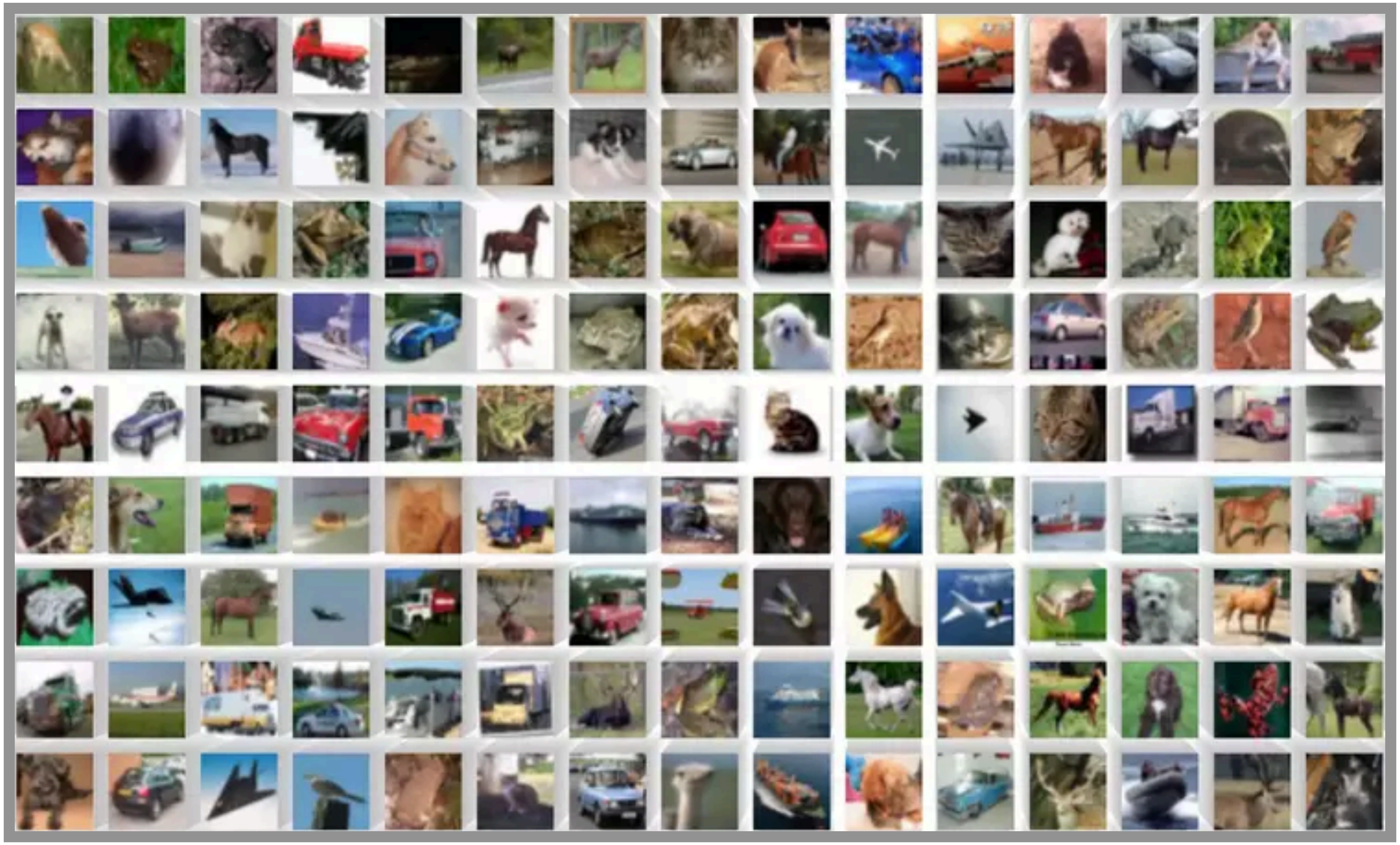
$C(x) = \{$  ,  $\}$

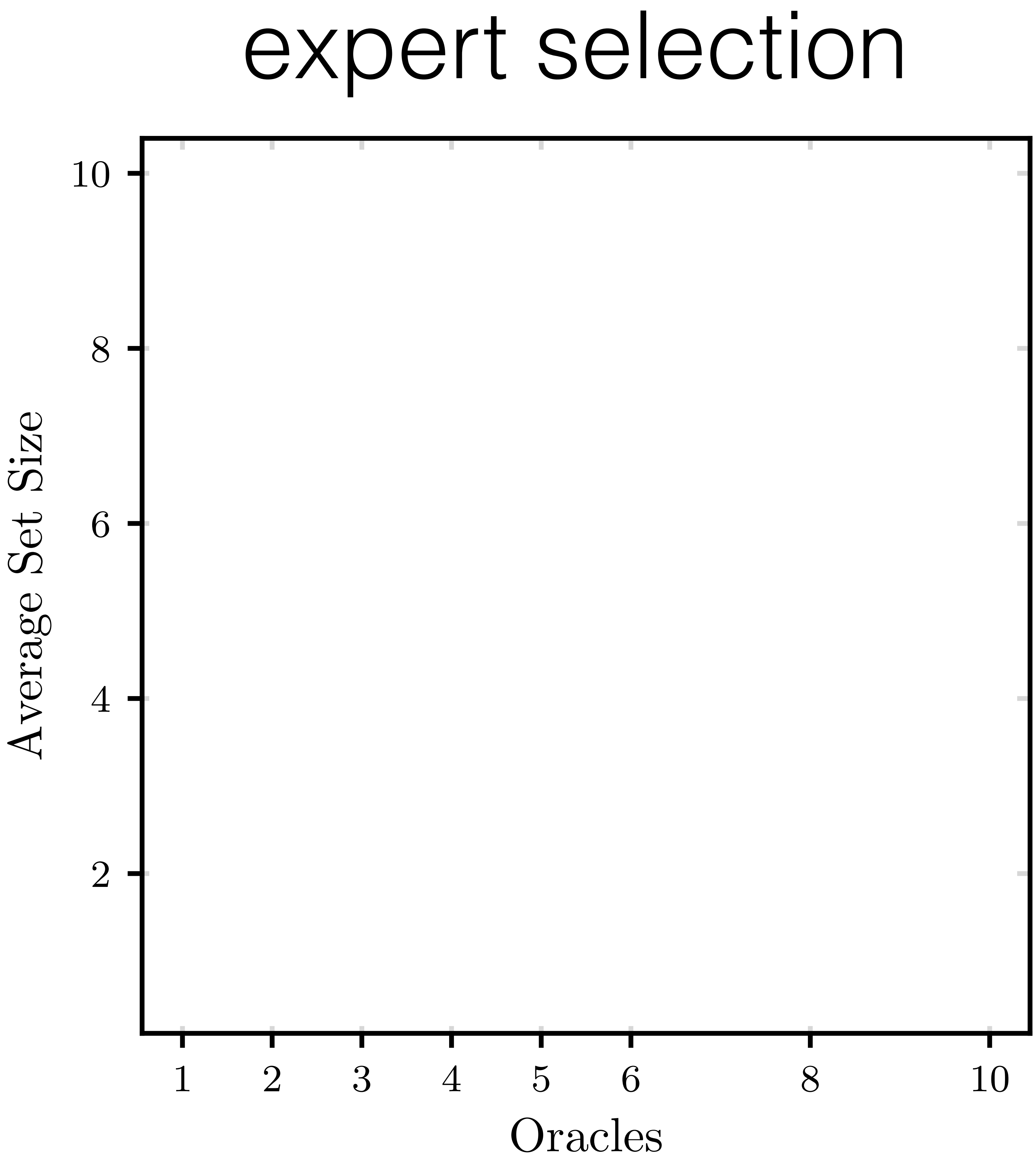
predictions: m_3 m_1

aggregated prediction: $\bar{m}$
(e.g. by voting)

expert selection

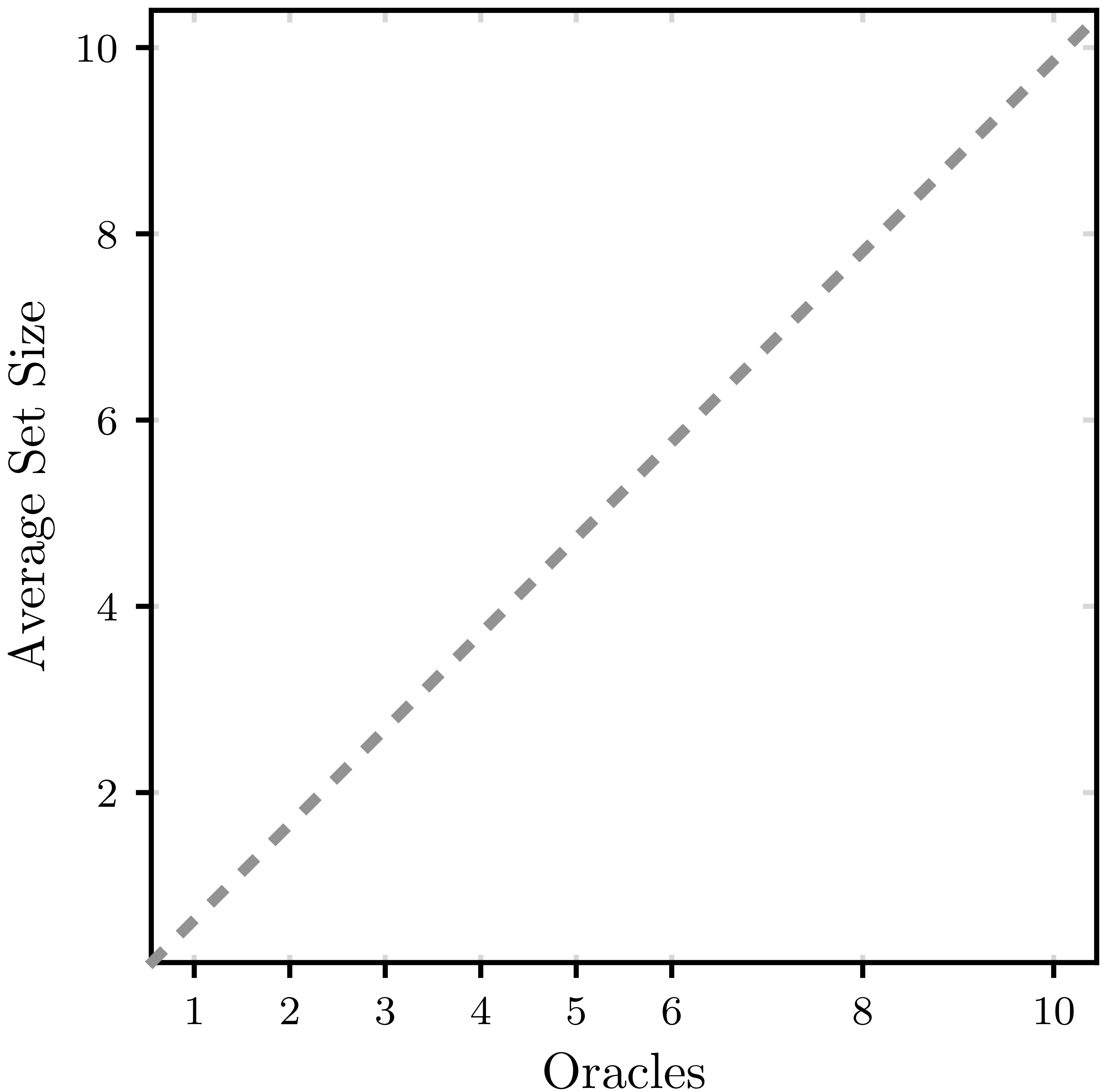
CIFAR-10



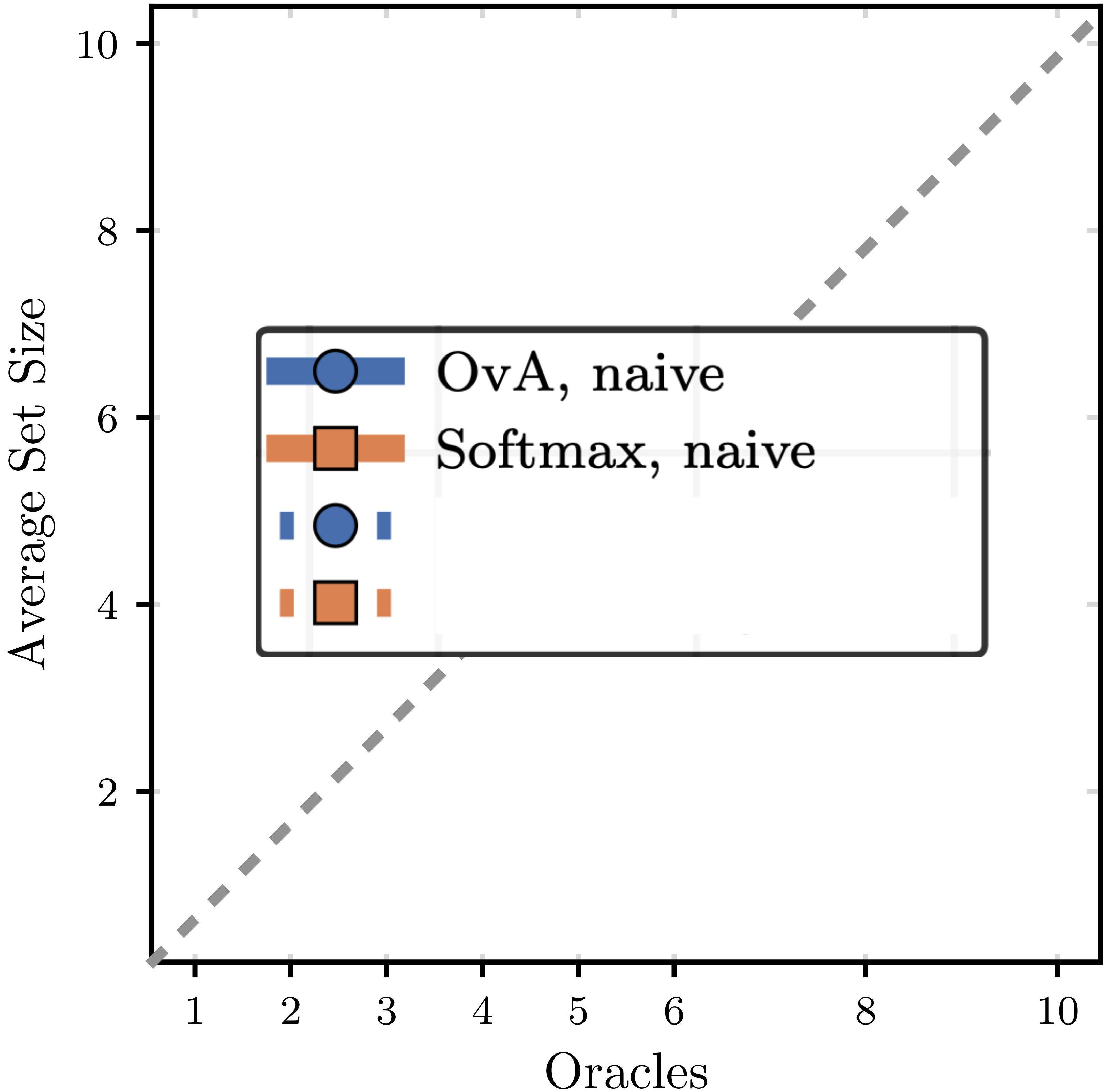


expert selection
Average Set Size
10
8
6
4
2
1
2
3
4
5
6
8
10
Oracles

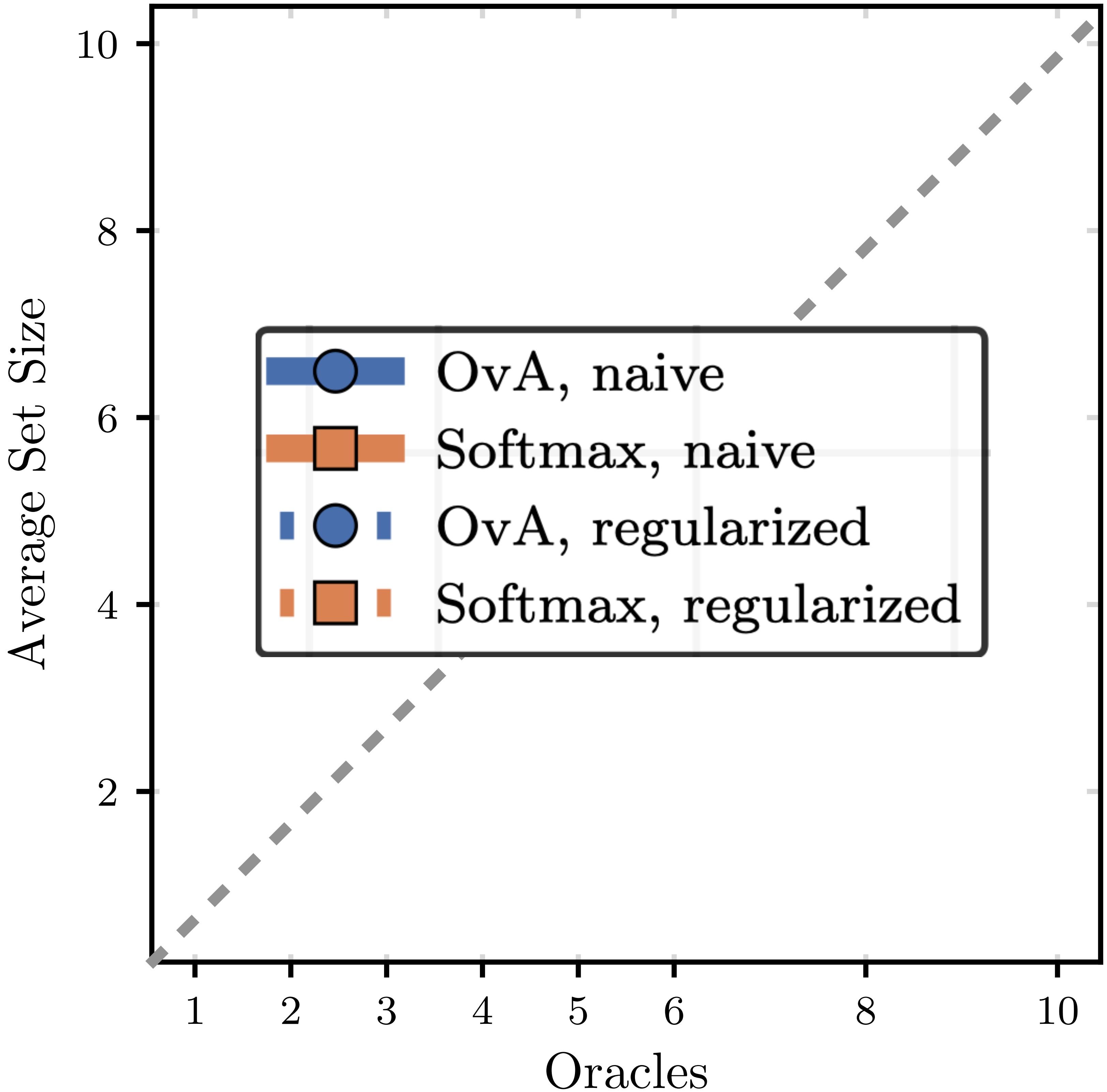
expert selection

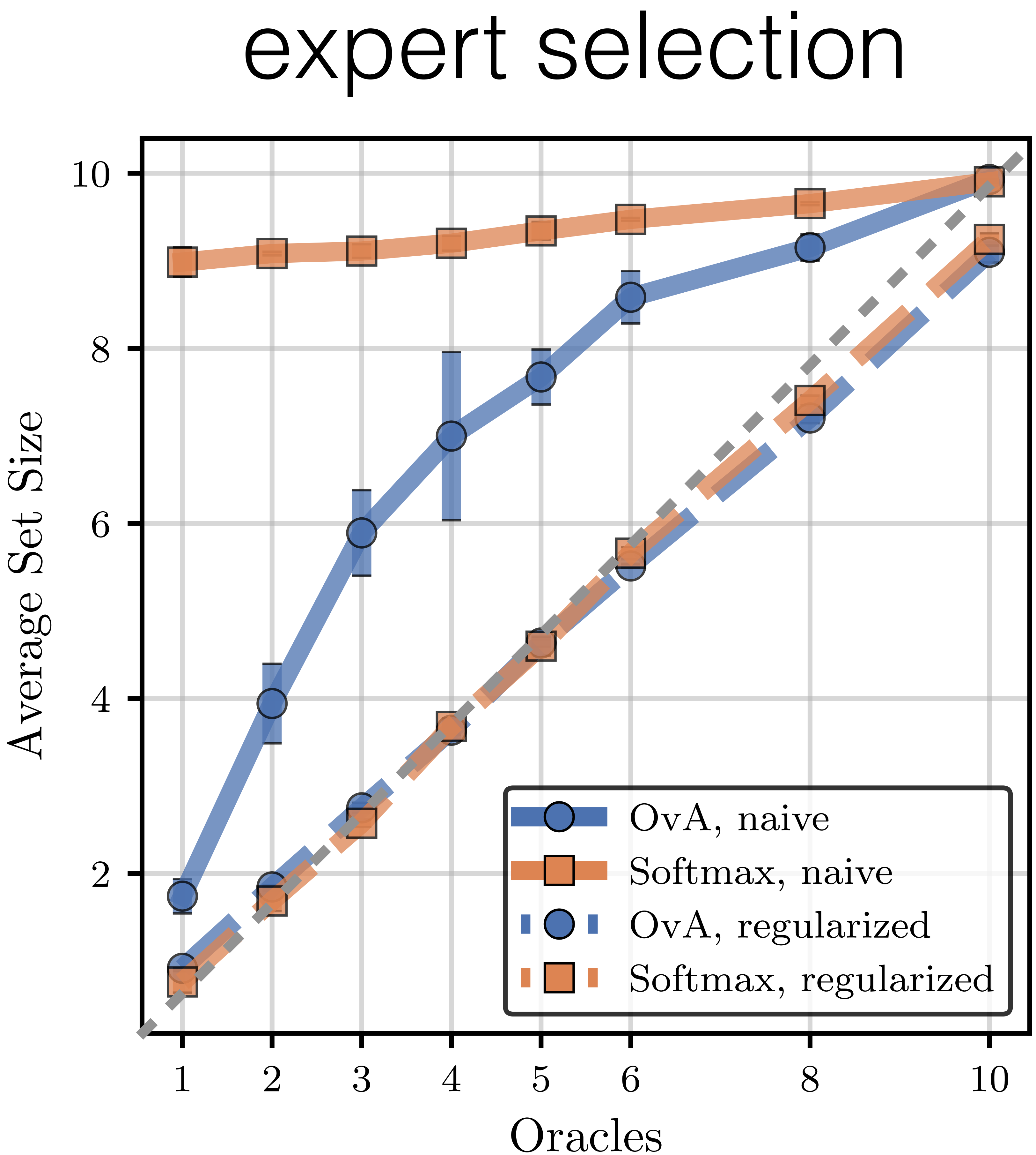


expert selection



expert selection

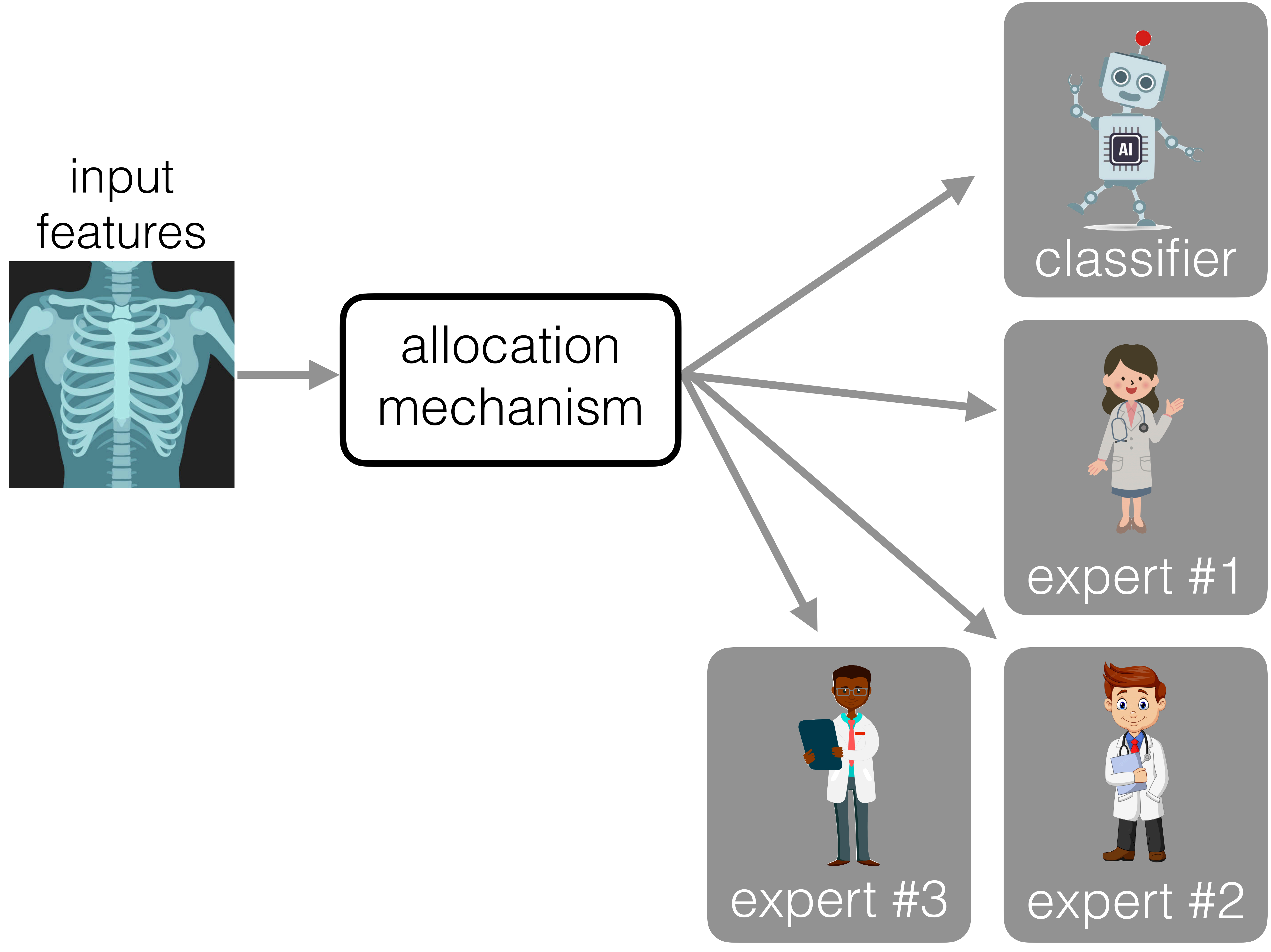


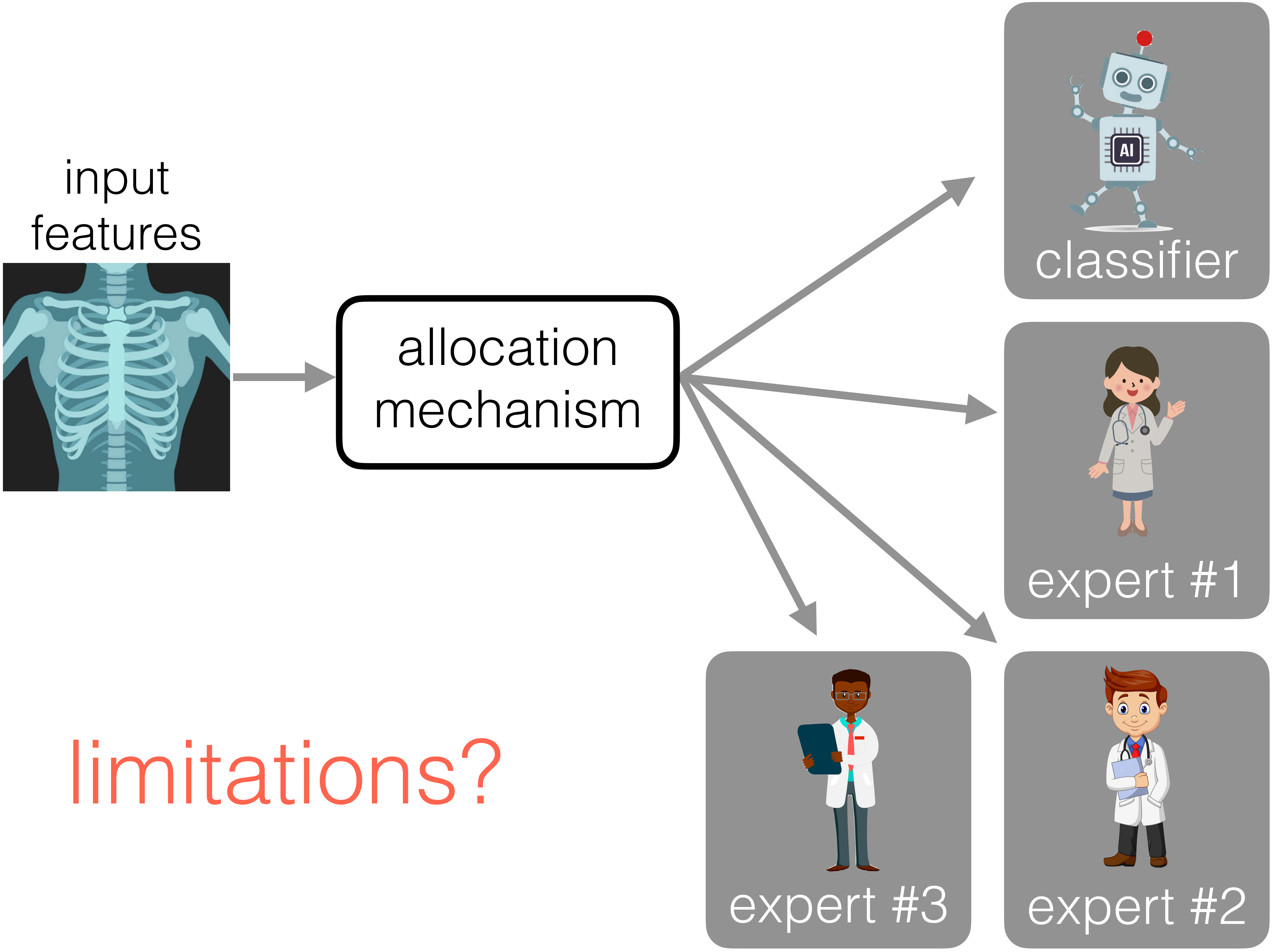


expert selection
Average Set Size
Oracles
OvA, naive
Softmax, naive
OvA, regularized
Softmax, regularized
10
8
6
4
2
1
2
3
4
5
6
8
10

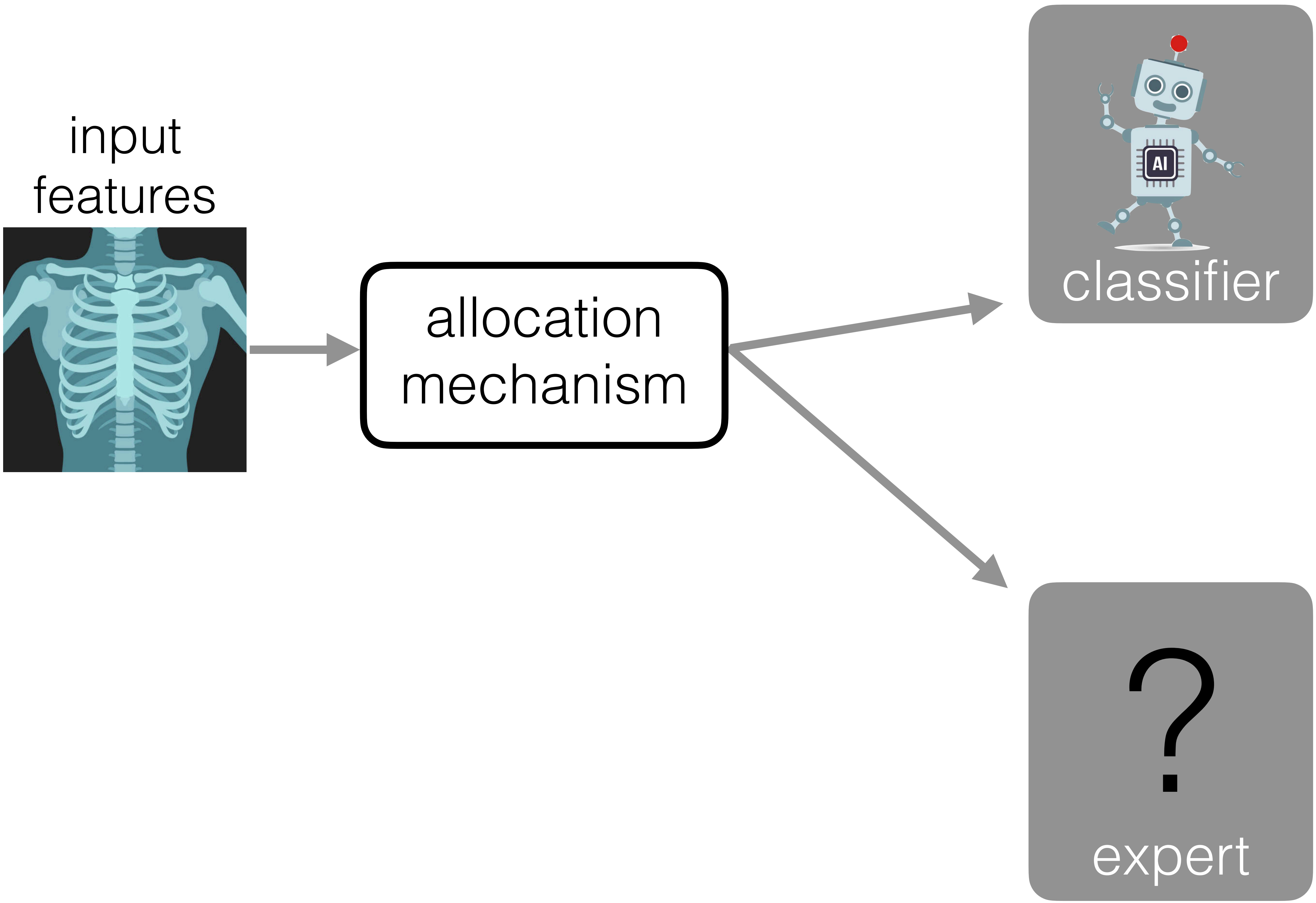
- ⊗ single expert
  - ⊗ softmax surrogate loss
  - ⊗ improving calibration via one-vs-all
- ⊗ **multiple experts**
  - ⊗ surrogate losses
  - ⊗ conformal sets of experts
- ⊗ population of experts
  - ⊗ surrogate losses
  - ⊗ meta-learning a rejector

- ⊗ single expert
  - ⊗ softmax surrogate loss
  - ⊗ improving calibration via one-vs-all
- ⊗ multiple experts
  - ⊗ surrogate losses
  - ⊗ conformal sets of experts
- ⊗ population of experts
  - ⊗ surrogate losses
  - ⊗ meta-learning a rejector

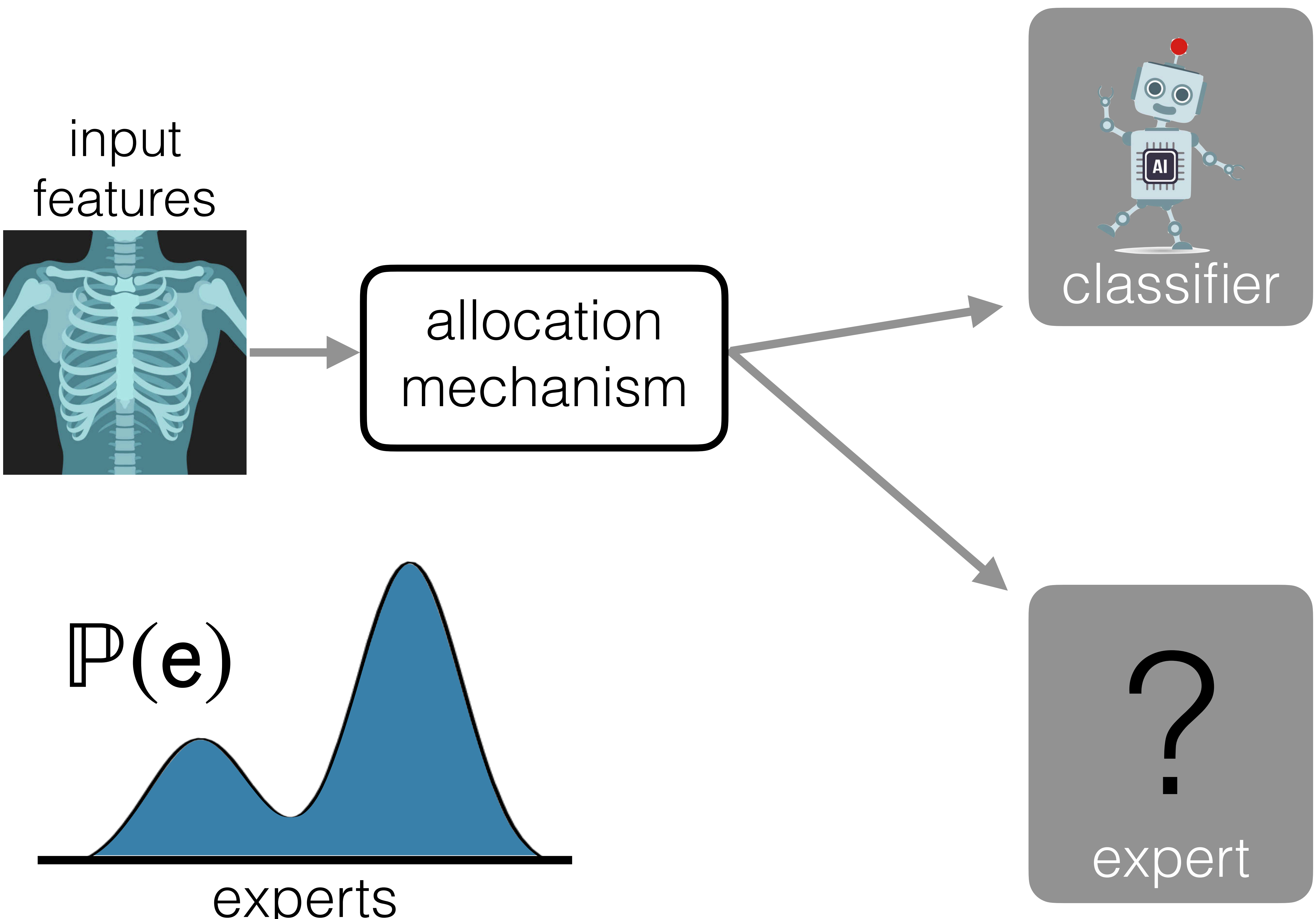

input
features
allocation
mechanism
classifier
AI
expert #1
expert #2
expert #3

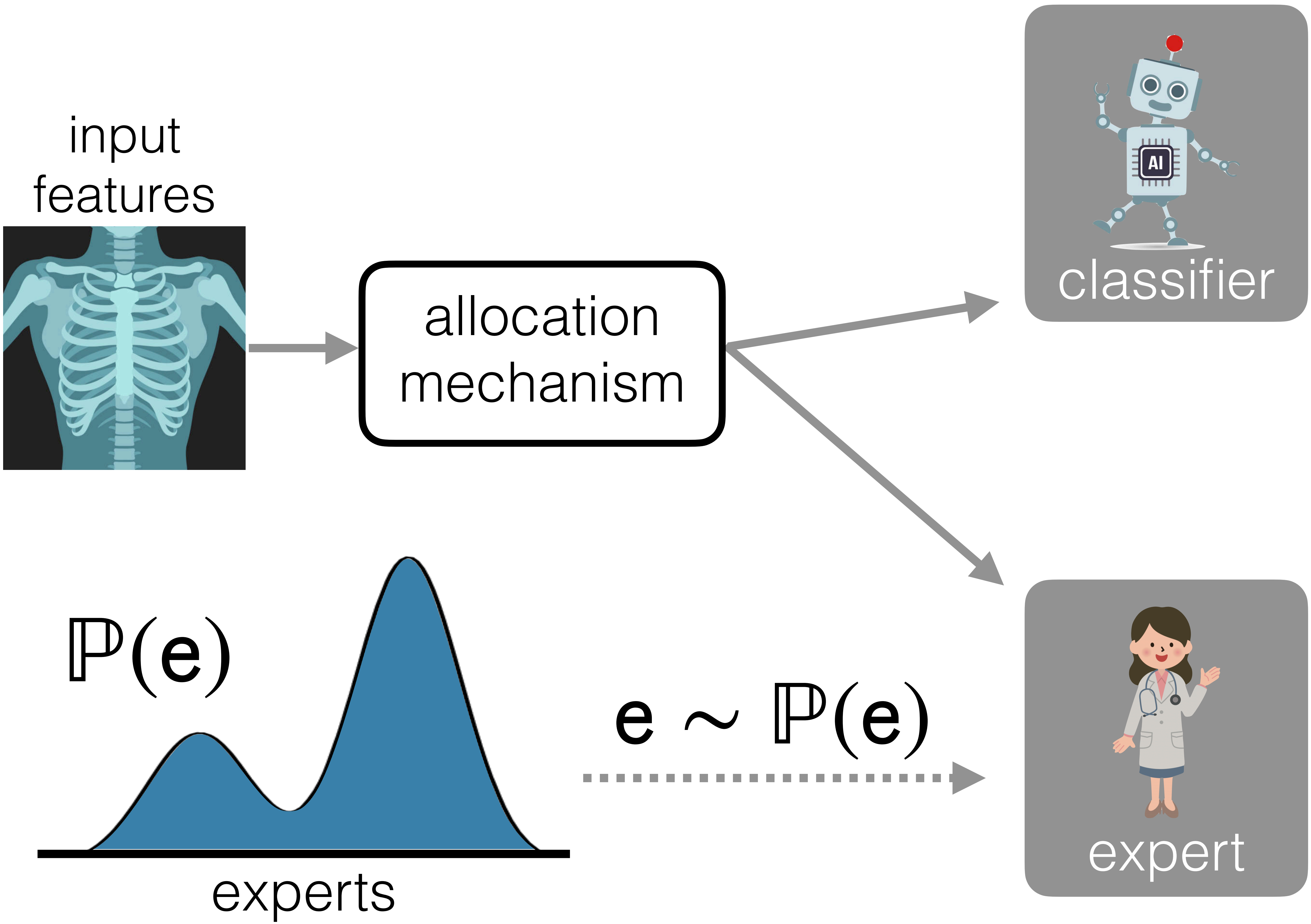


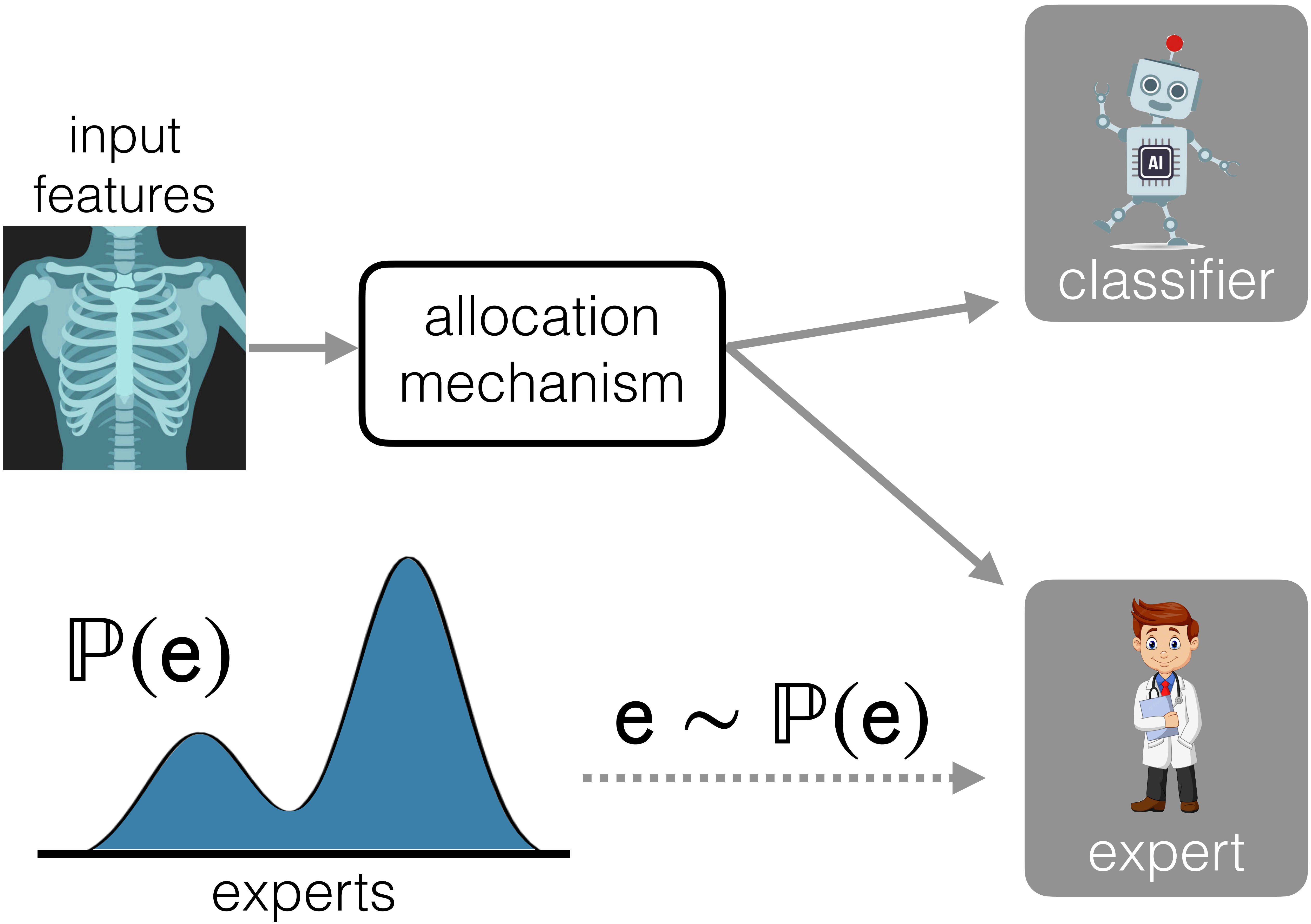
limitations?

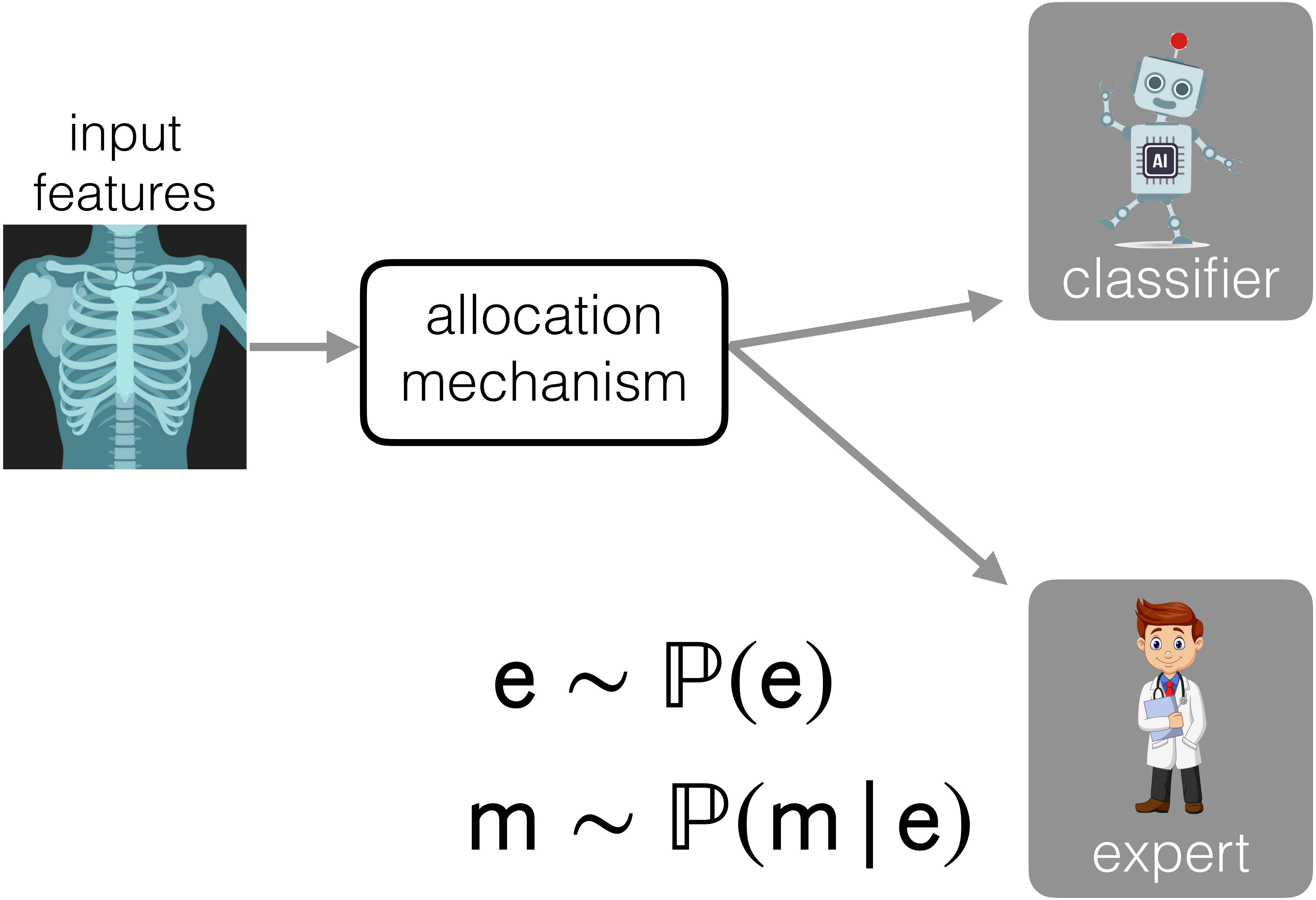


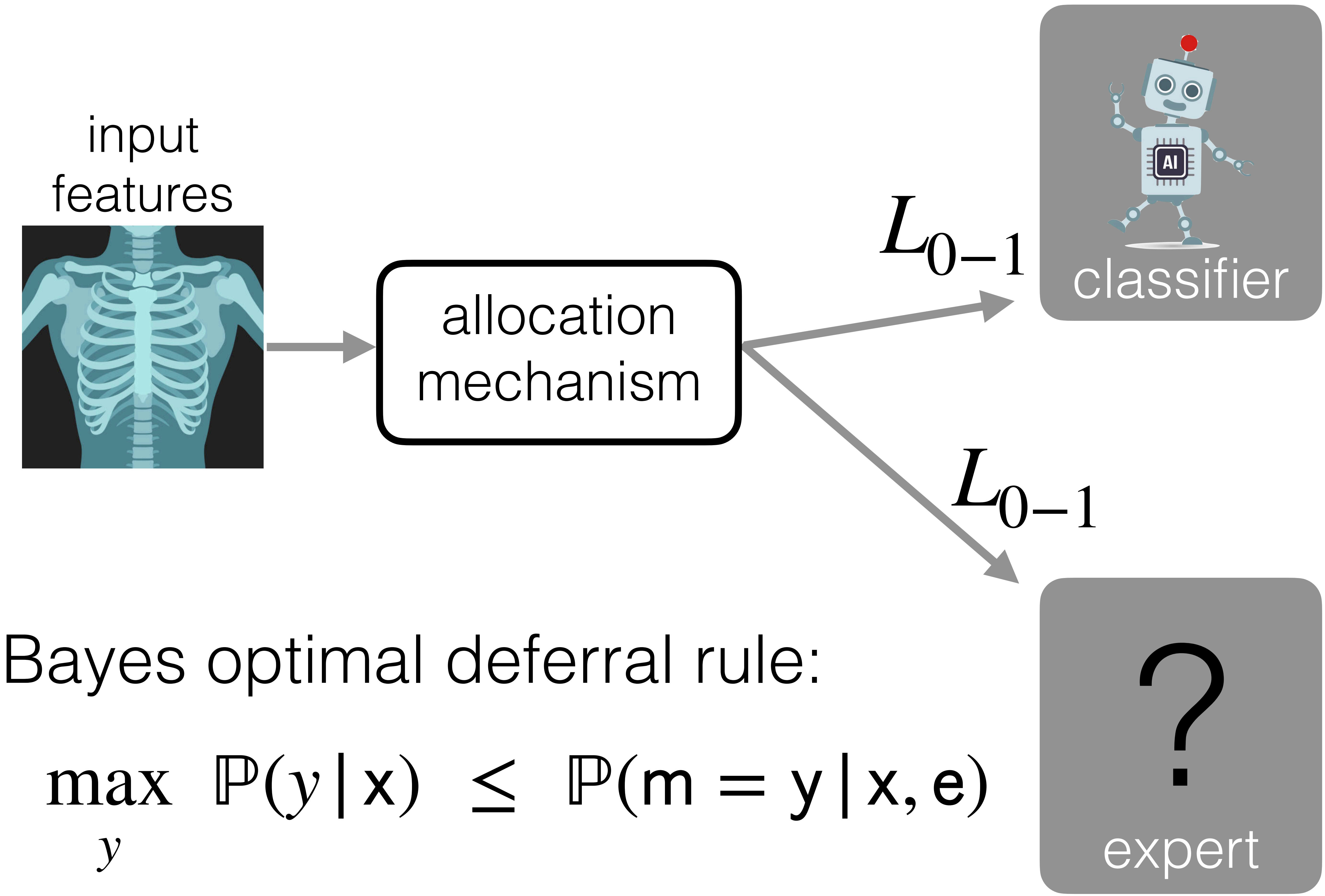
input
features
allocation
mechanism
AI
classifier
?
expert

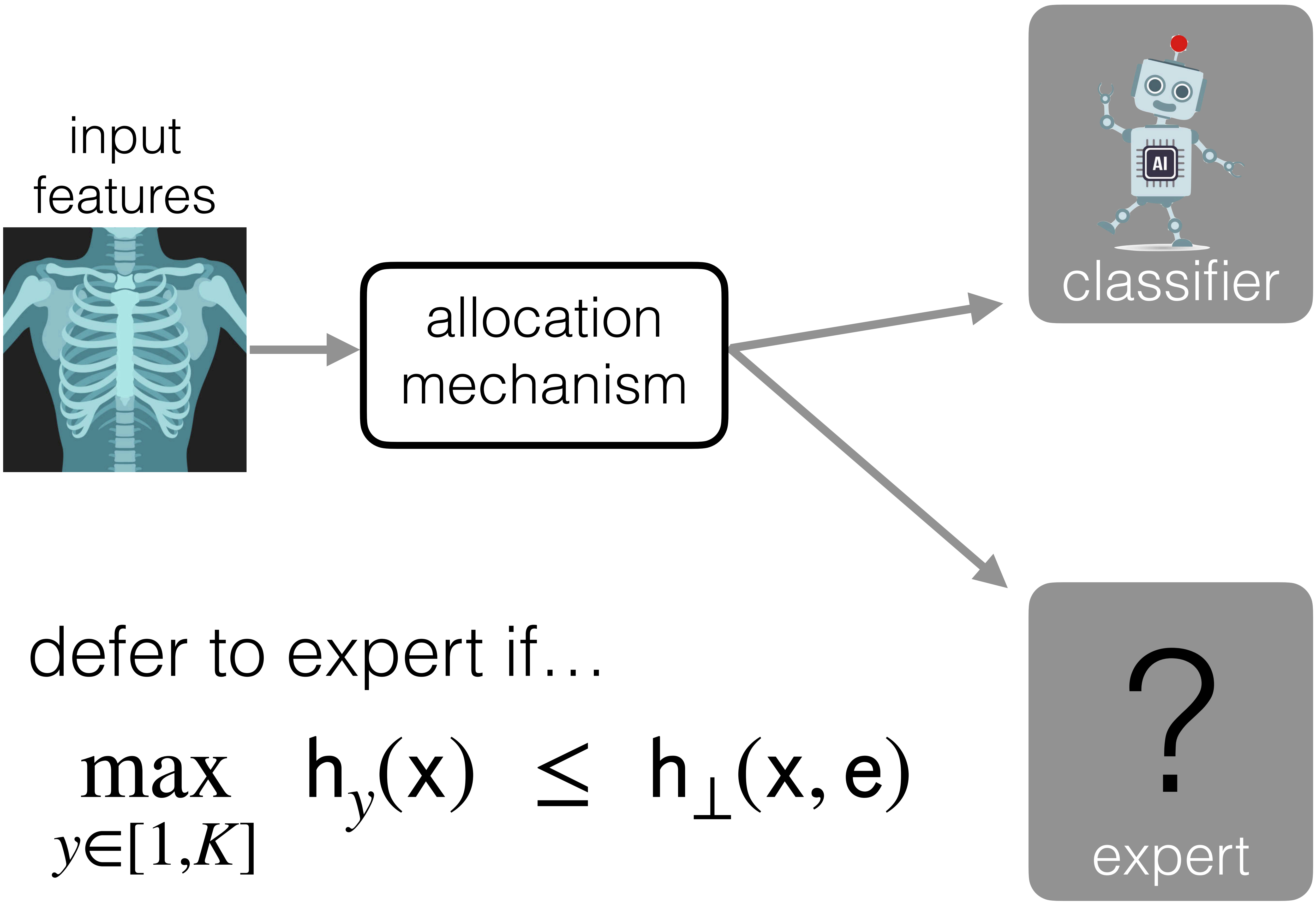

input
features
allocation
mechanism
classifier
AI
?
expert
$\mathbb{P}(\mathbf{e})$
experts

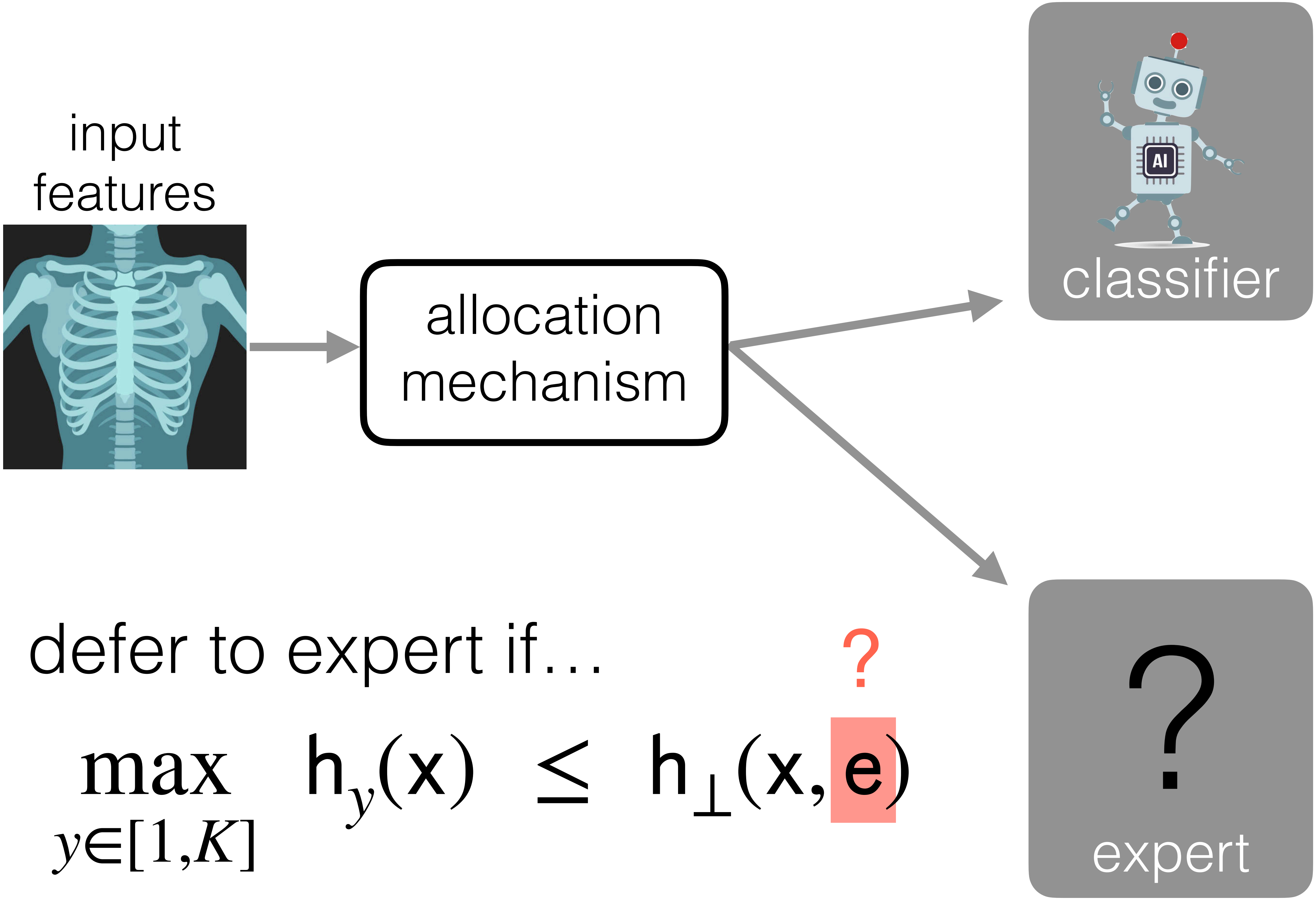

input
features
allocation
mechanism
classifier
AI
expert
$\mathbb{P}(\mathbf{e})$
$\mathbf{e} \sim \mathbb{P}(\mathbf{e})$
experts


input
features
allocation
mechanism
classifier
AI
$\mathbb{P}(\mathbf{e})$
$\mathbf{e} \sim \mathbb{P}(\mathbf{e})$
expert
experts


input
features
allocation
mechanism
classifier
AI
expert
$\mathbf{e} \sim \mathbb{P}(\mathbf{e})$
$\mathbf{m} \sim \mathbb{P}(\mathbf{m} | \mathbf{e})$


input
features
allocation
mechanism
L_{0-1}
classifier
AI
L_{0-1}
?
expert
Bayes optimal deferral rule:
$\max_{y} \mathbb{P}(y \,|\, \mathsf{x}) \leq \mathbb{P}(\mathsf{m} = \mathsf{y} \,|\, \mathsf{x}, \mathsf{e})$

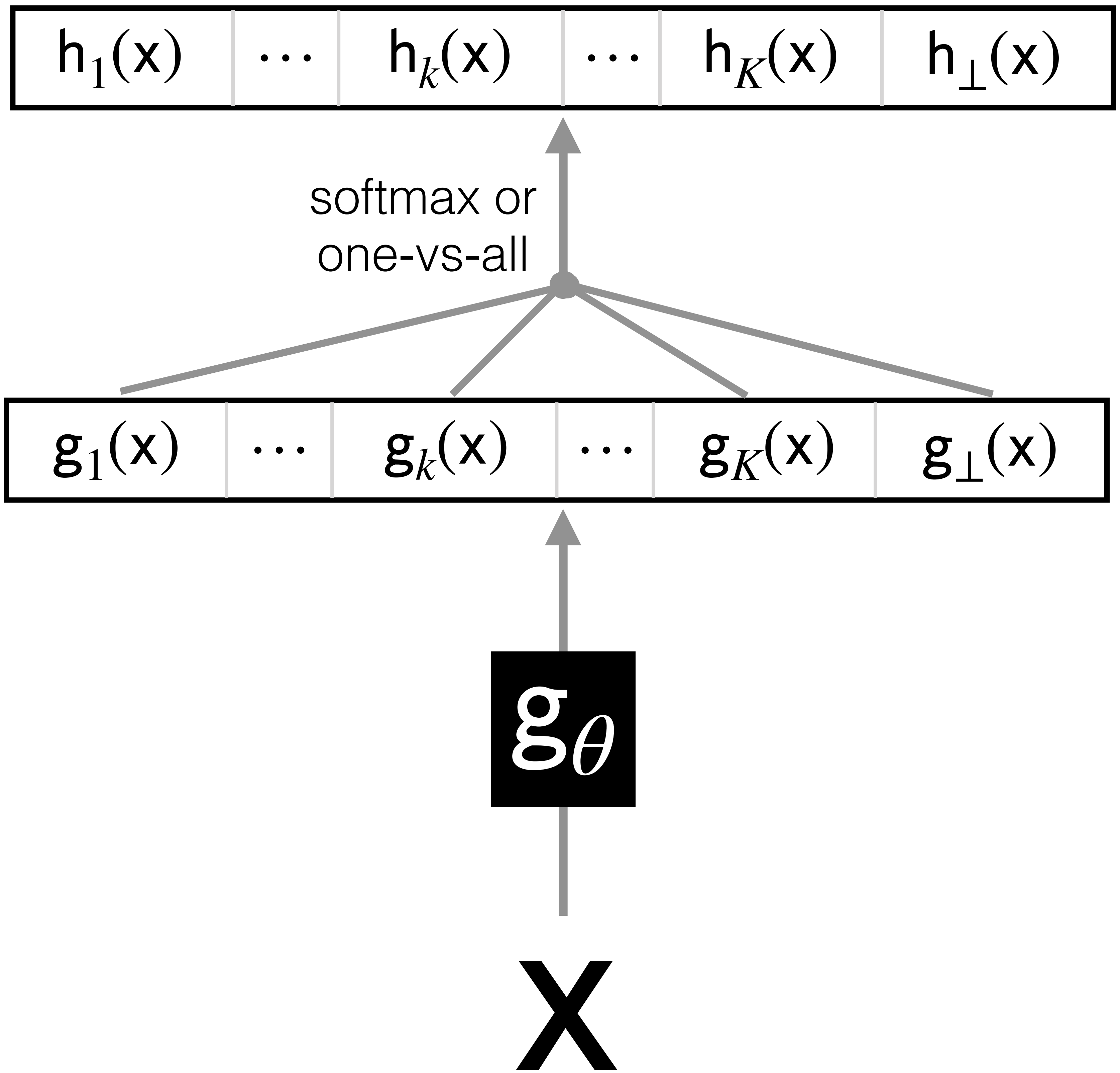

input features
allocation mechanism
classifier
?
expert
defer to expert if…
$\max_{y \in [1,K]} \mathrm{h}_y(\mathbf{x}) \leq \mathrm{h}_\perp(\mathbf{x}, \mathbf{e})$


input features
allocation mechanism
classifier
AI
expert
?
defer to expert if…
$\max_{y \in [1,K]} h_y(x) \leq h_\perp(x, e)$
?

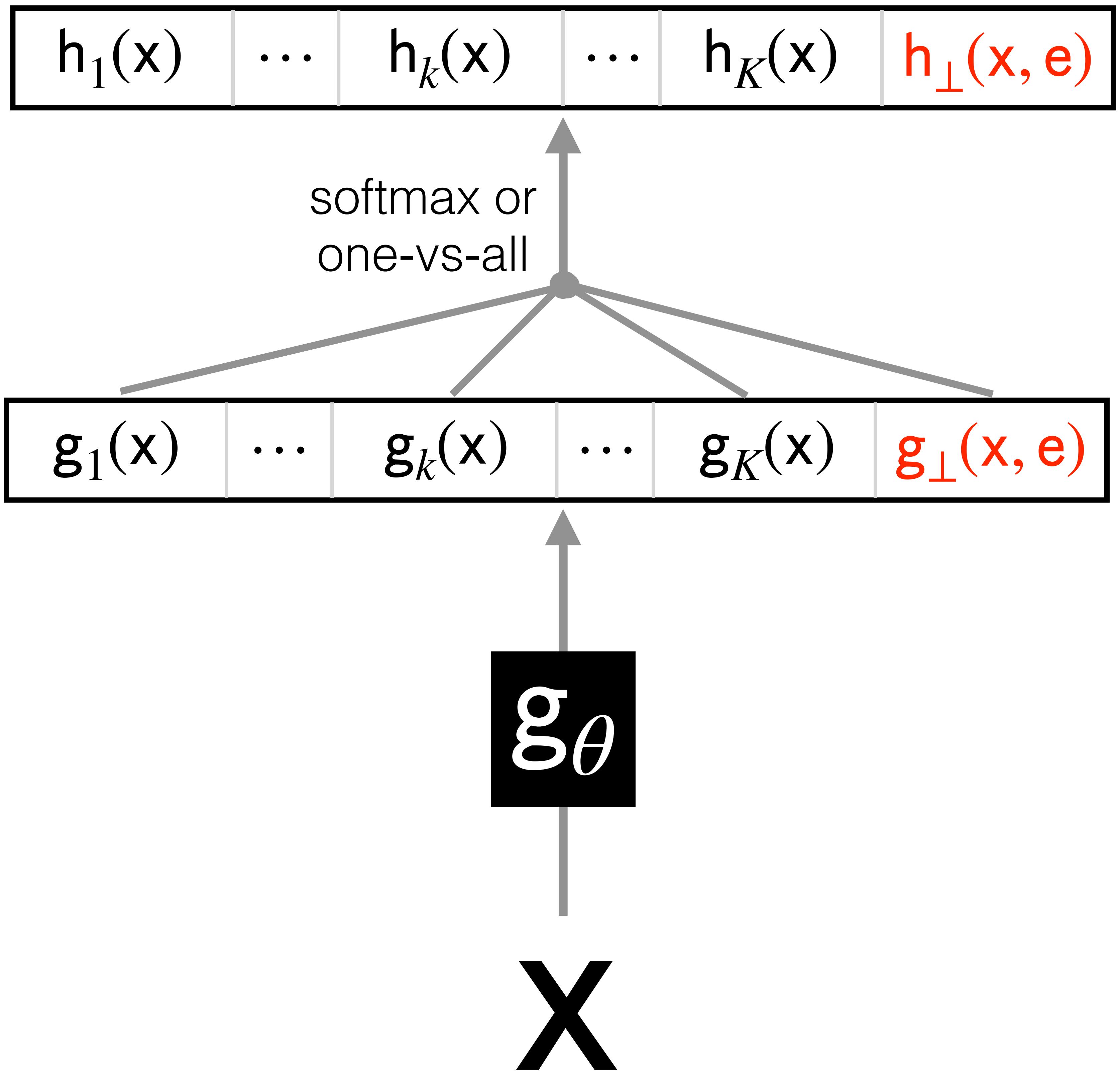
- ⊗ single expert
  - ⊗ softmax surrogate loss
  - ⊗ improving calibration via one-vs-all
- ⊗ multiple experts
  - ⊗ surrogate losses
  - ⊗ conformal sets of experts
- ⊗ population of experts
  - ⊗ surrogate losses
  - ⊗ meta-learning a rejector

- ⊗ single expert
  - ⊗ softmax surrogate loss
  - ⊗ improving calibration via one-vs-all
- ⊗ multiple experts
  - ⊗ surrogate losses
  - ⊗ conformal sets of experts
- ⊗ **population of experts**
  - ⊗ surrogate losses
  - ⊗ meta-learning a rejector

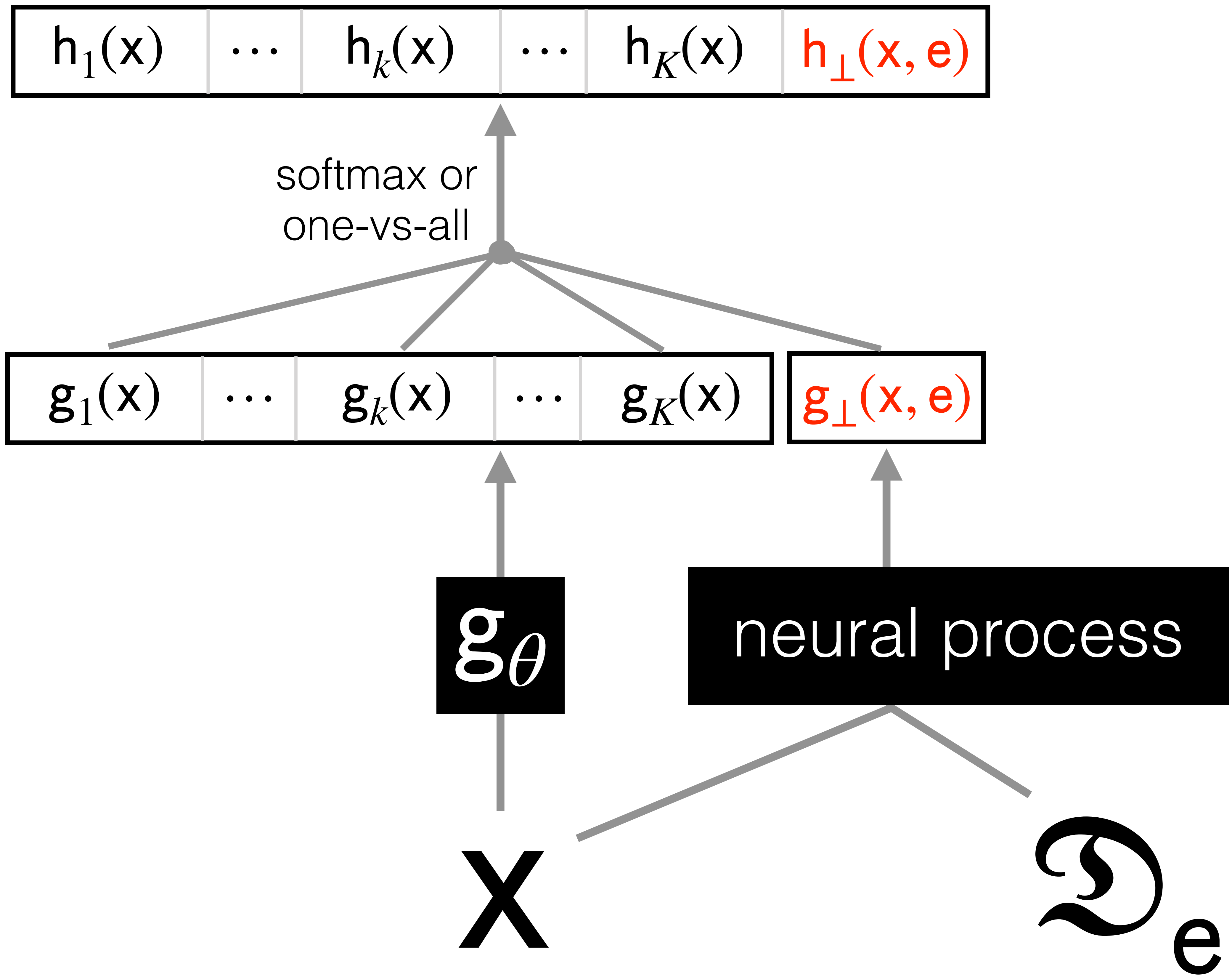
meta-learning implementation



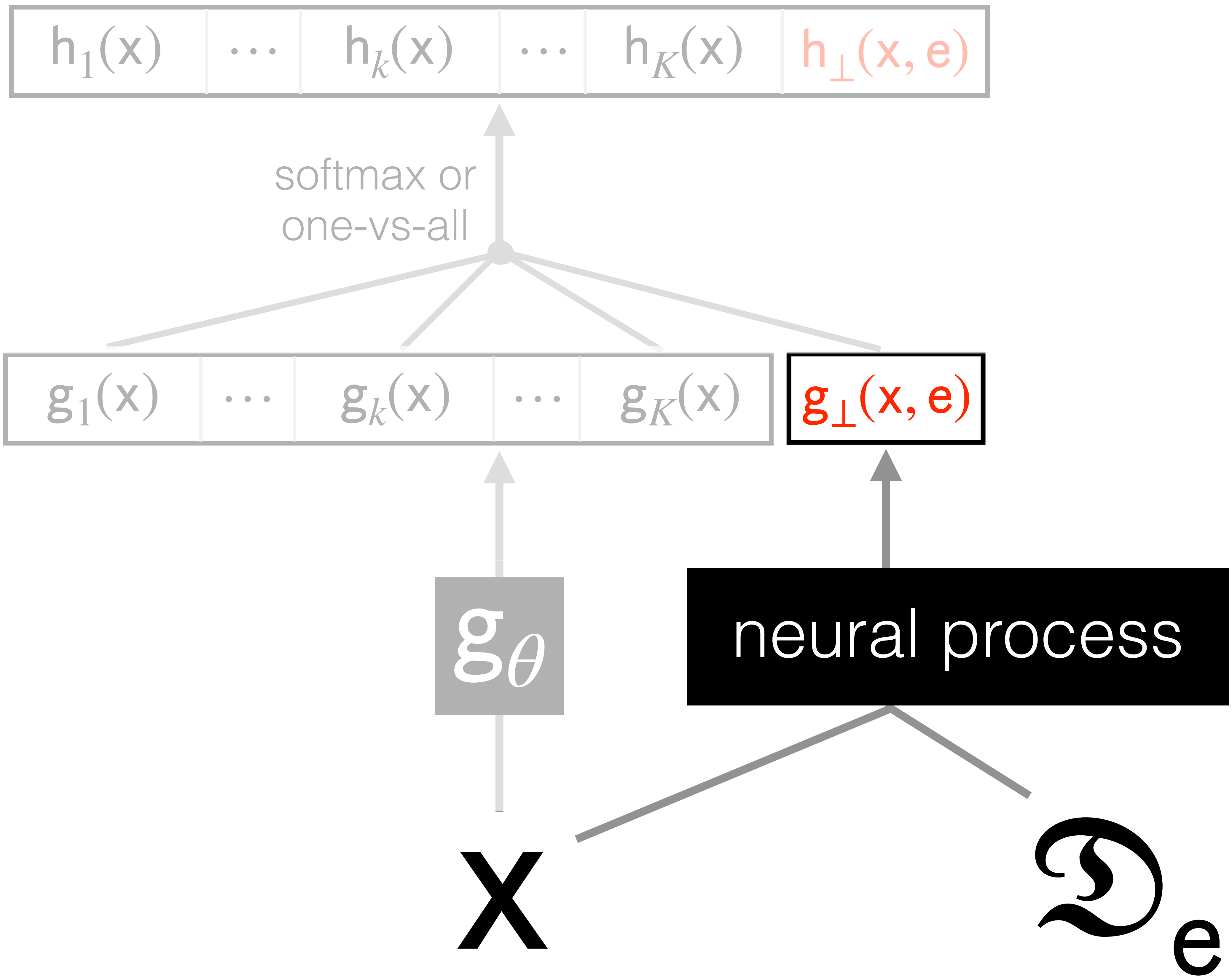
meta-learning implementation



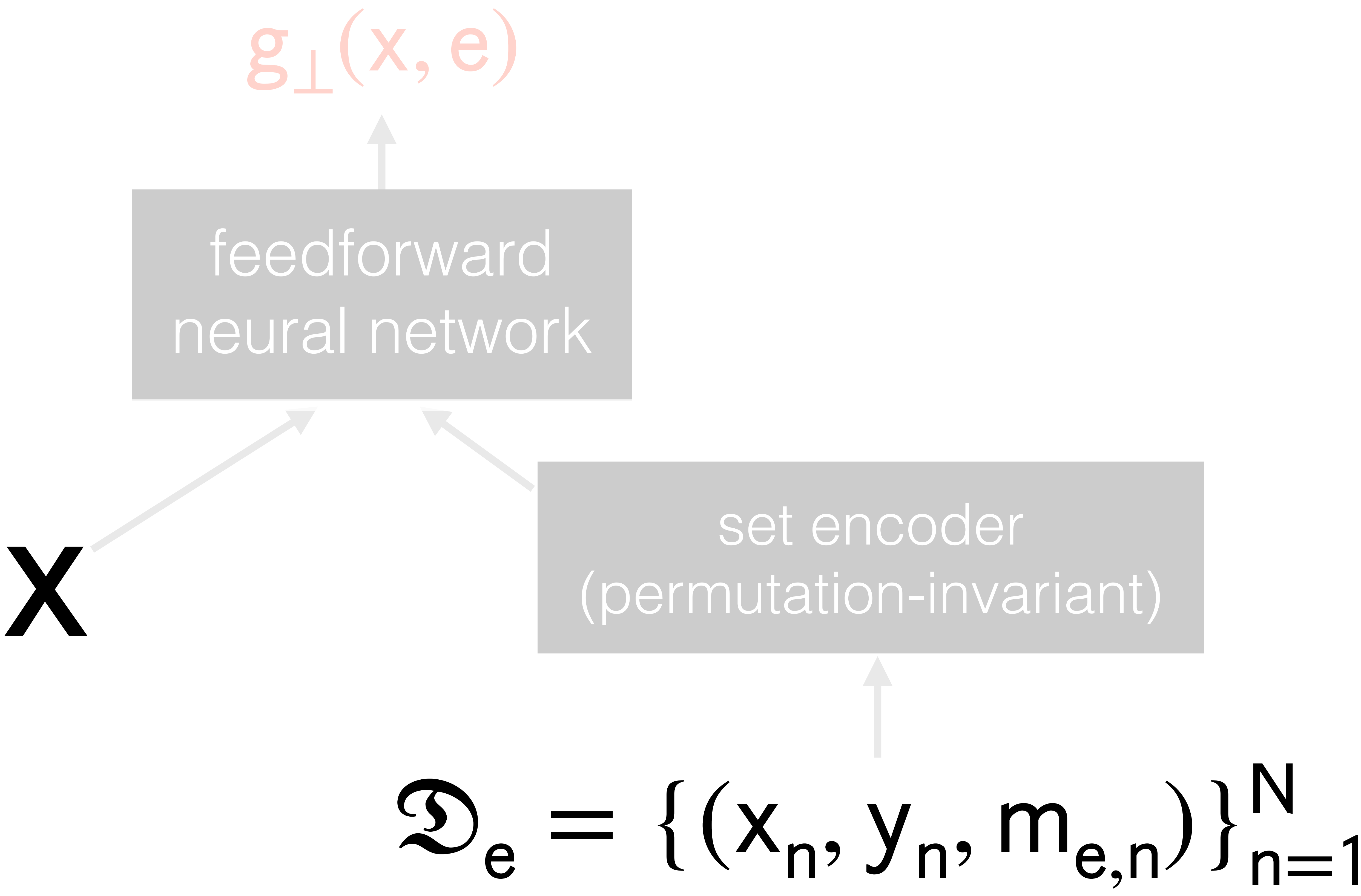
meta-learning implementation



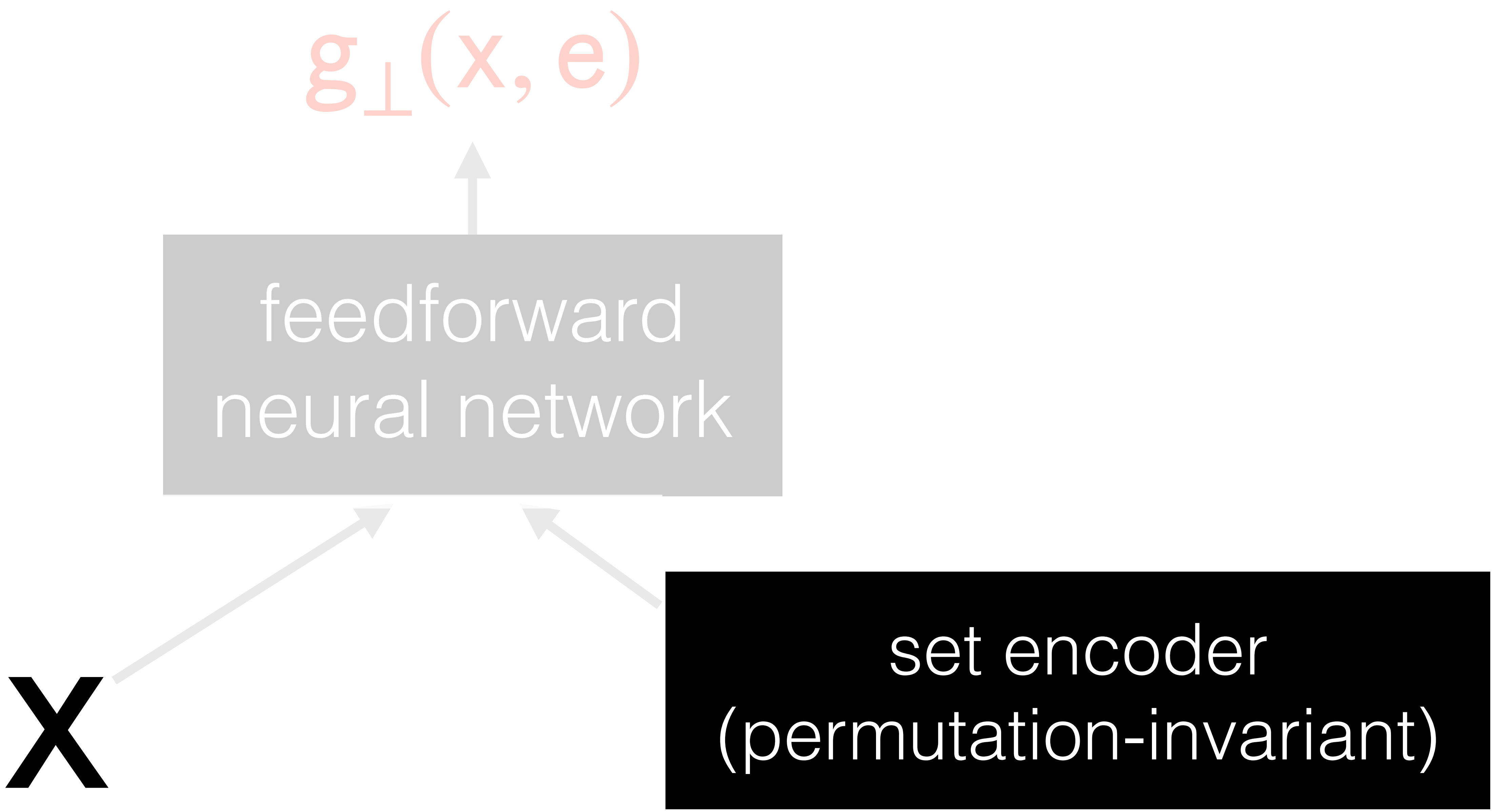
meta-learning implementation



neural process rejector

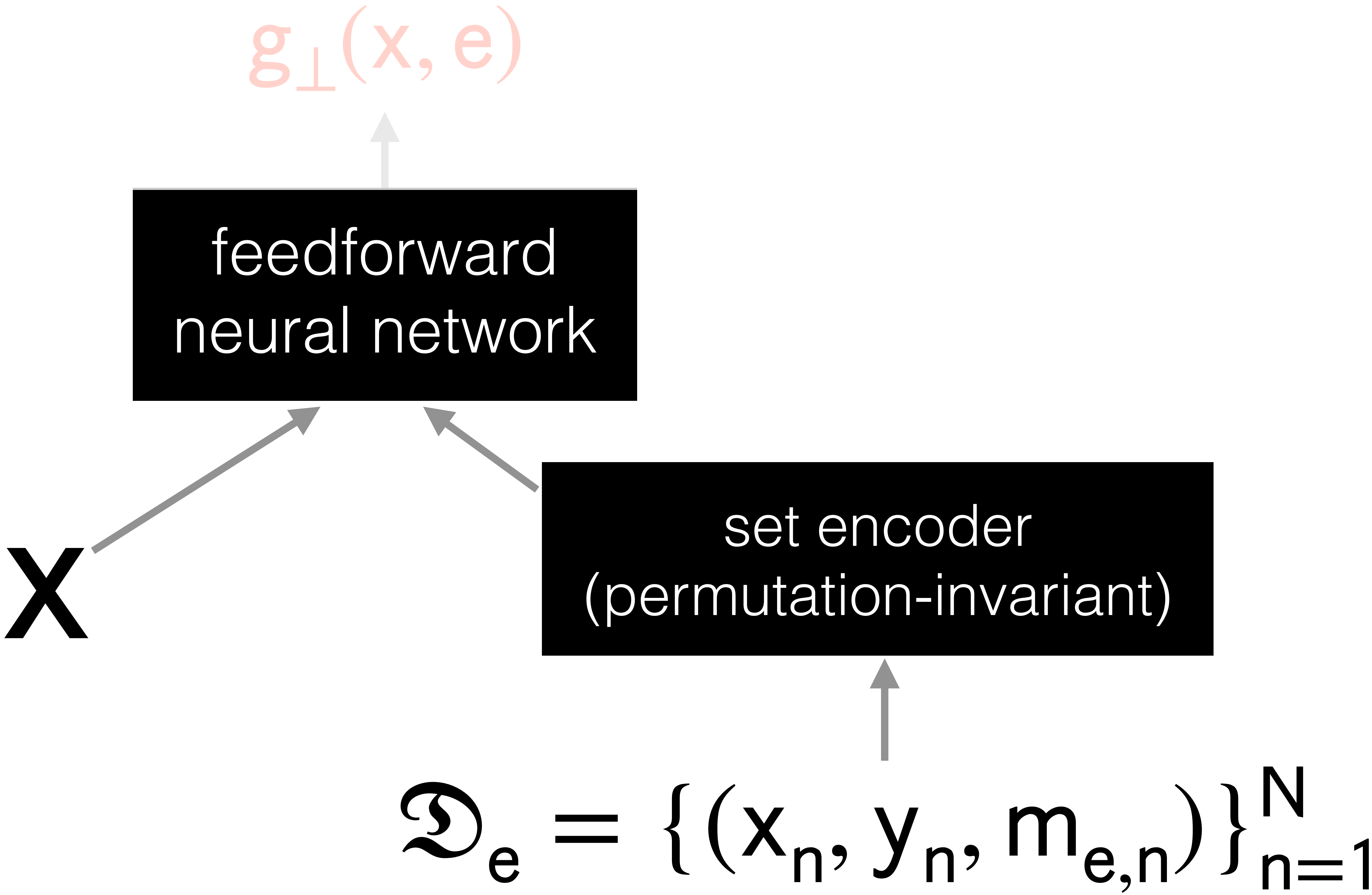


neural process rejector

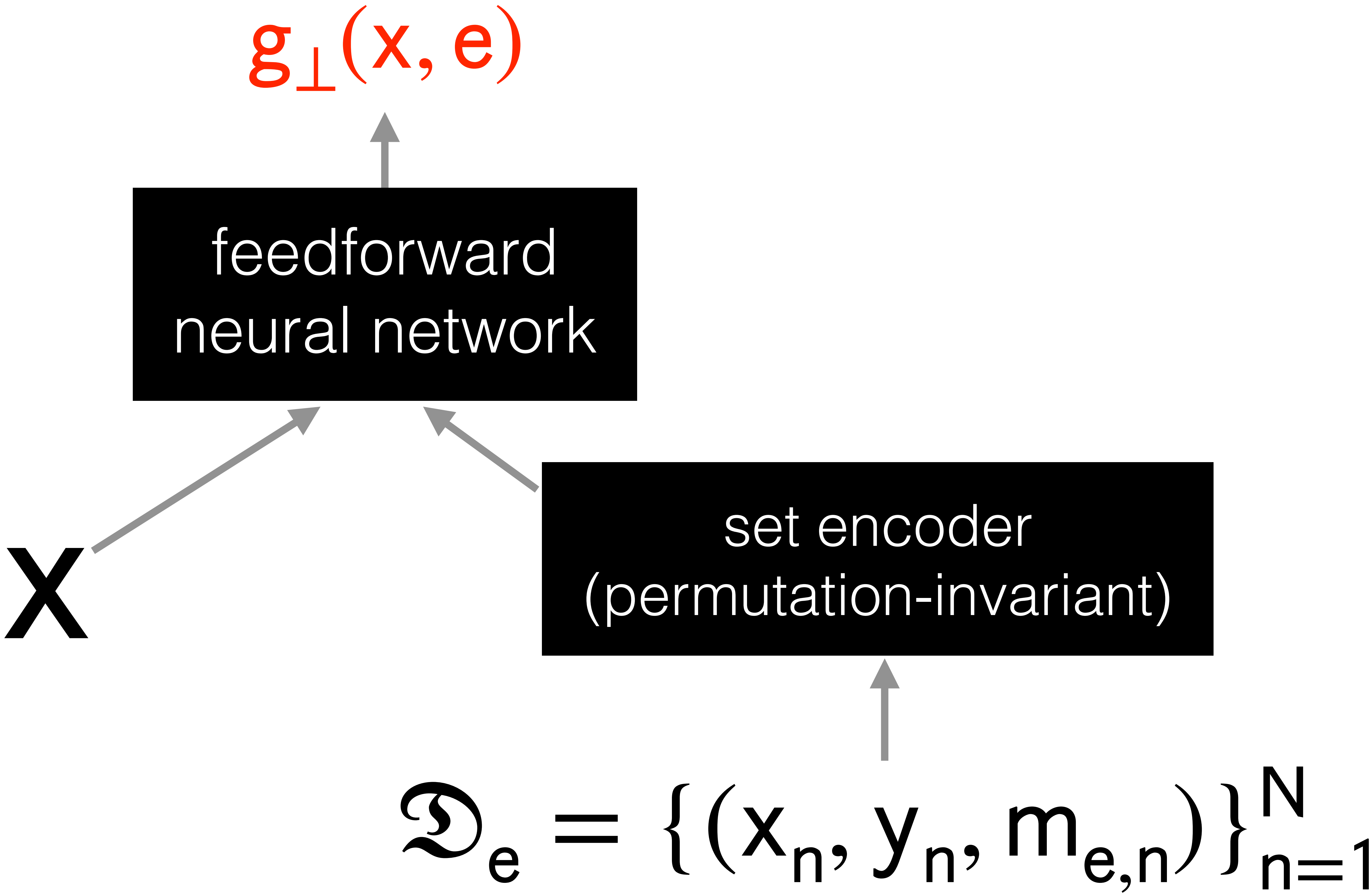


$$\mathfrak{D}_e = \{(x_n, y_n, m_{e,n})\}_{n=1}^{N}$$

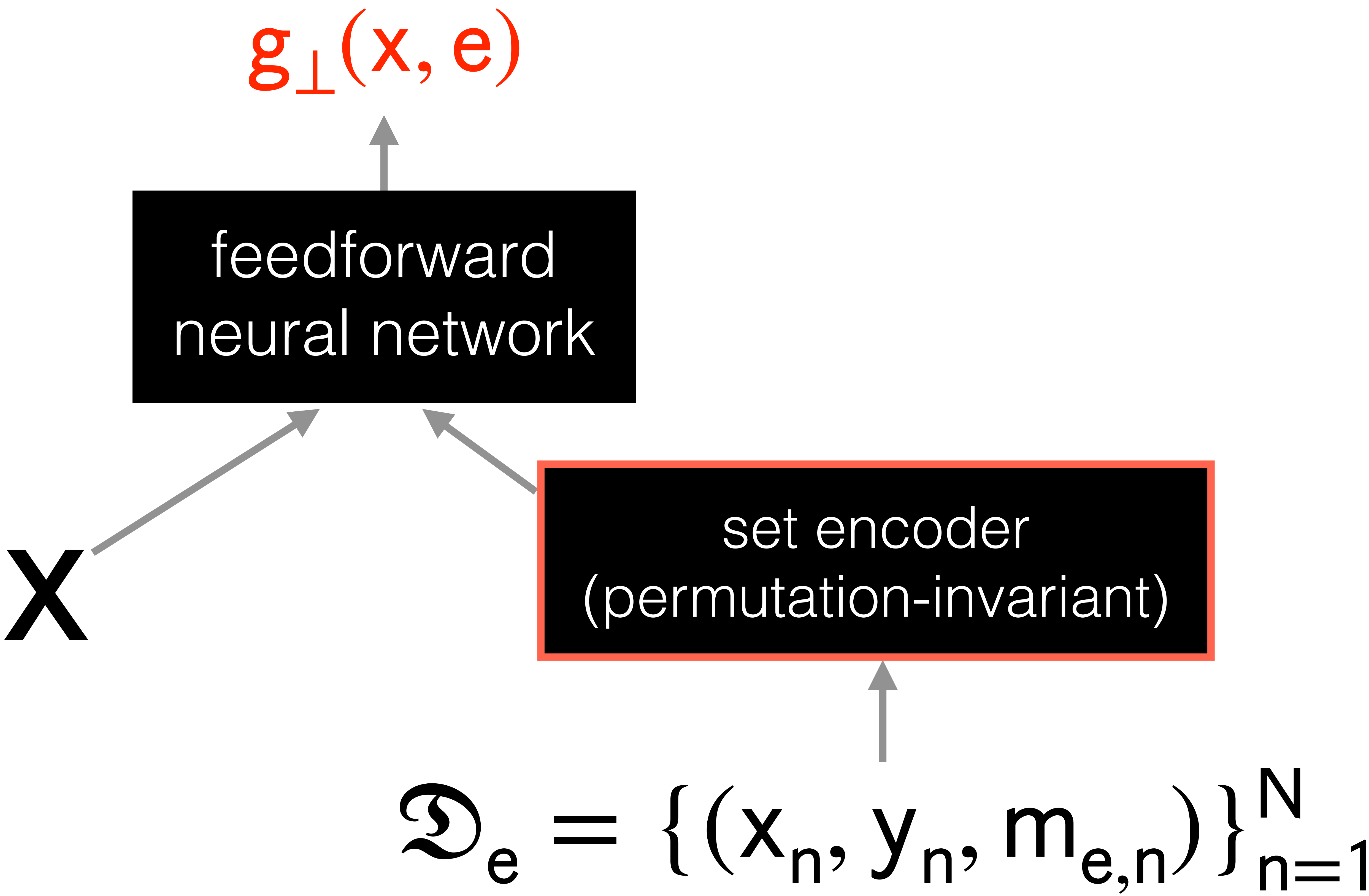
neural process rejector



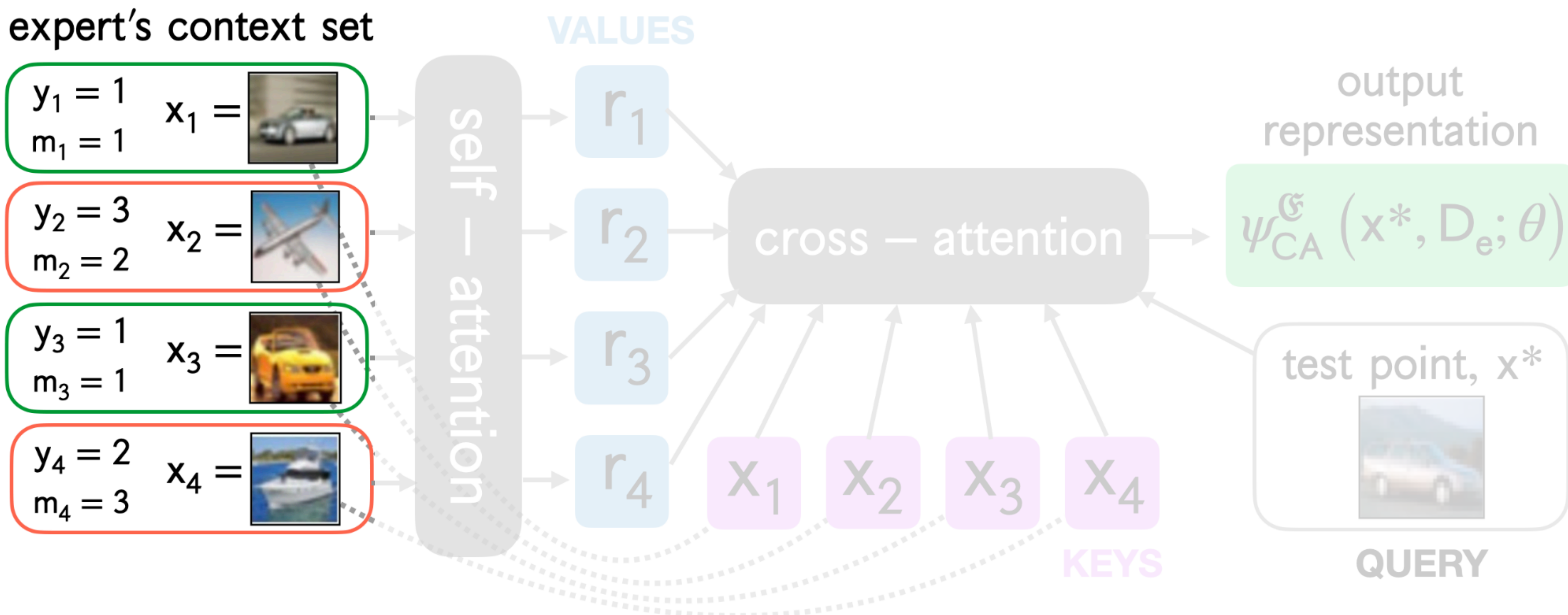
neural process rejector



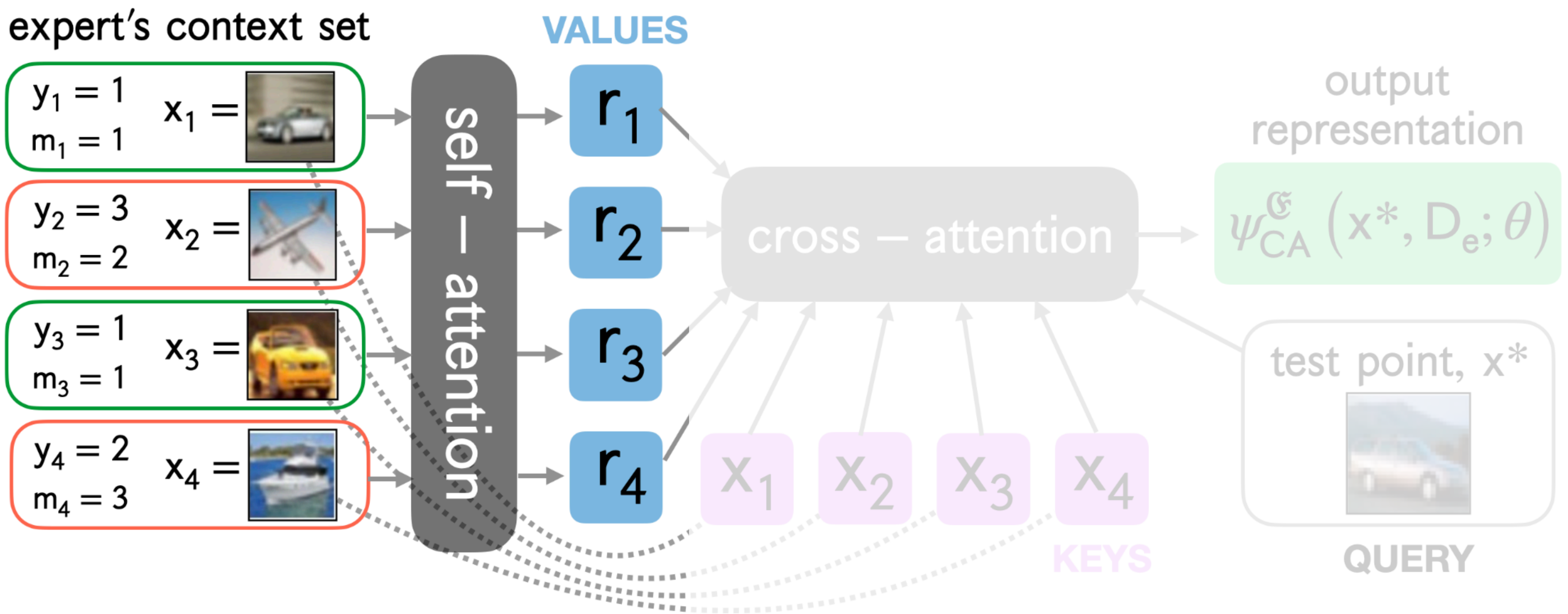
neural process rejector



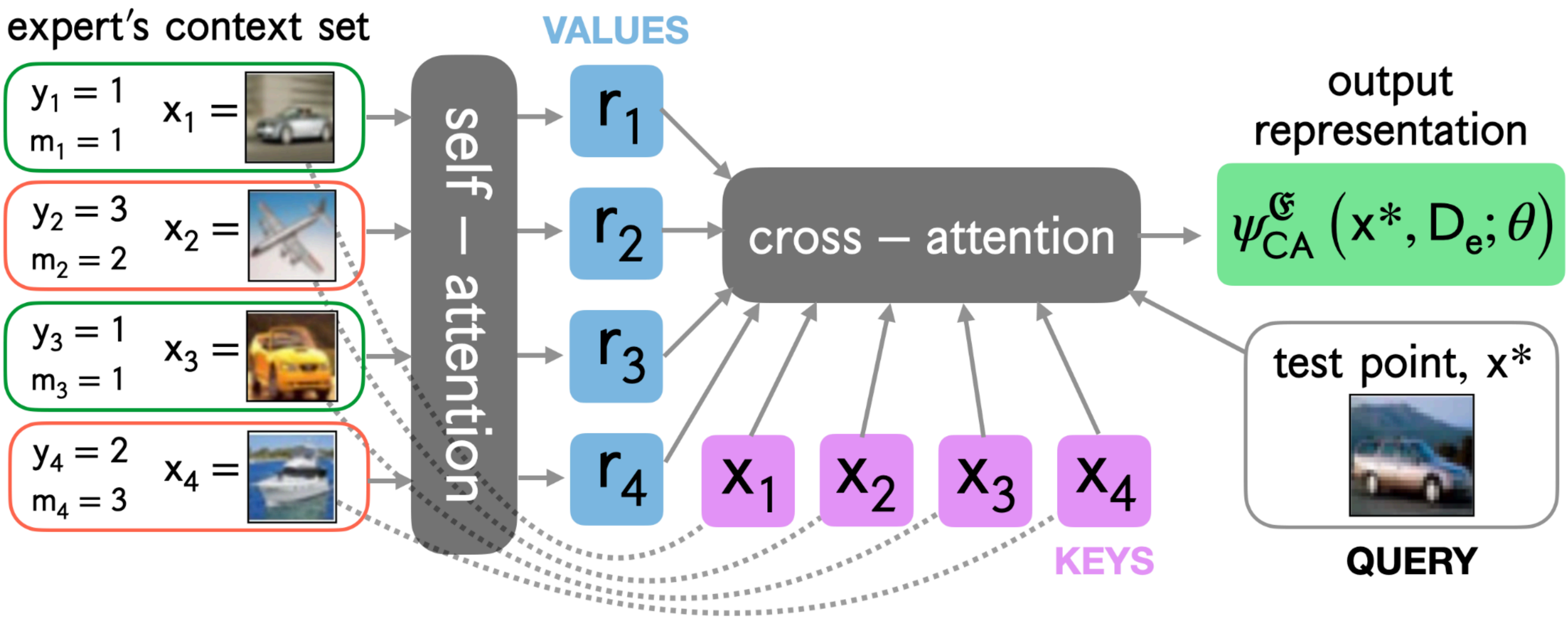
neural process rejector



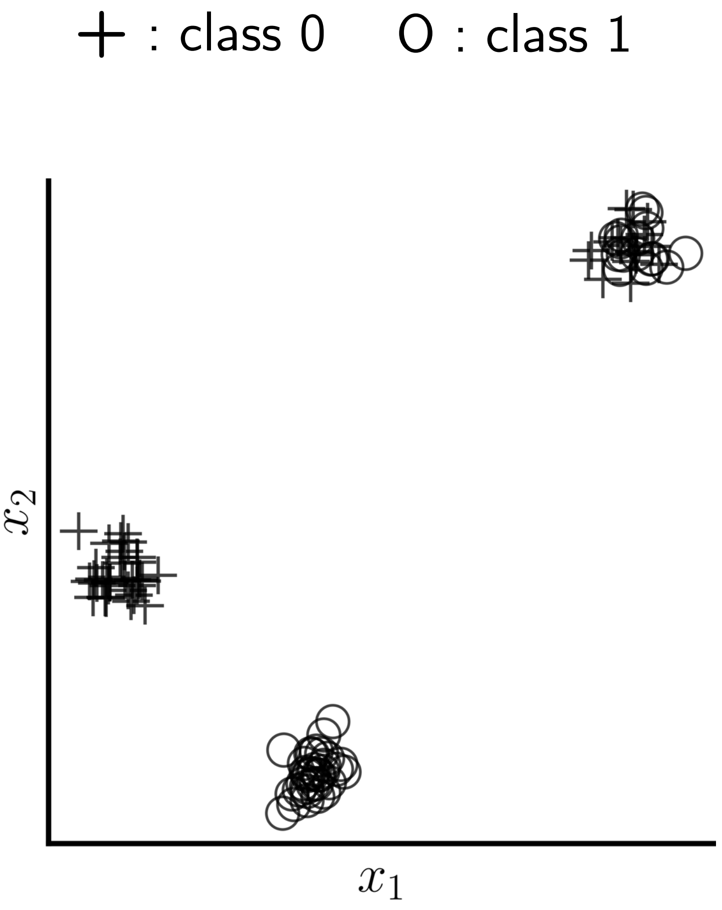
neural process rejector



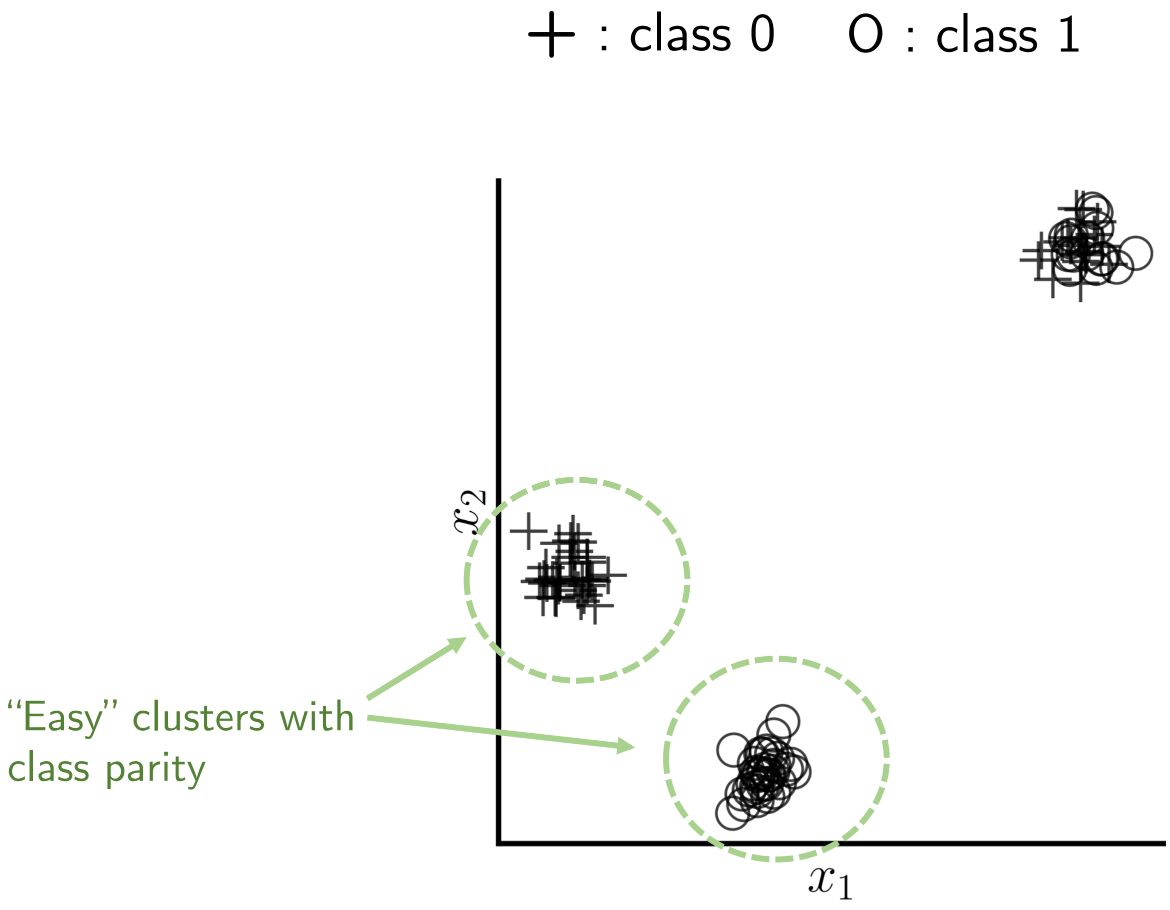
neural process rejector



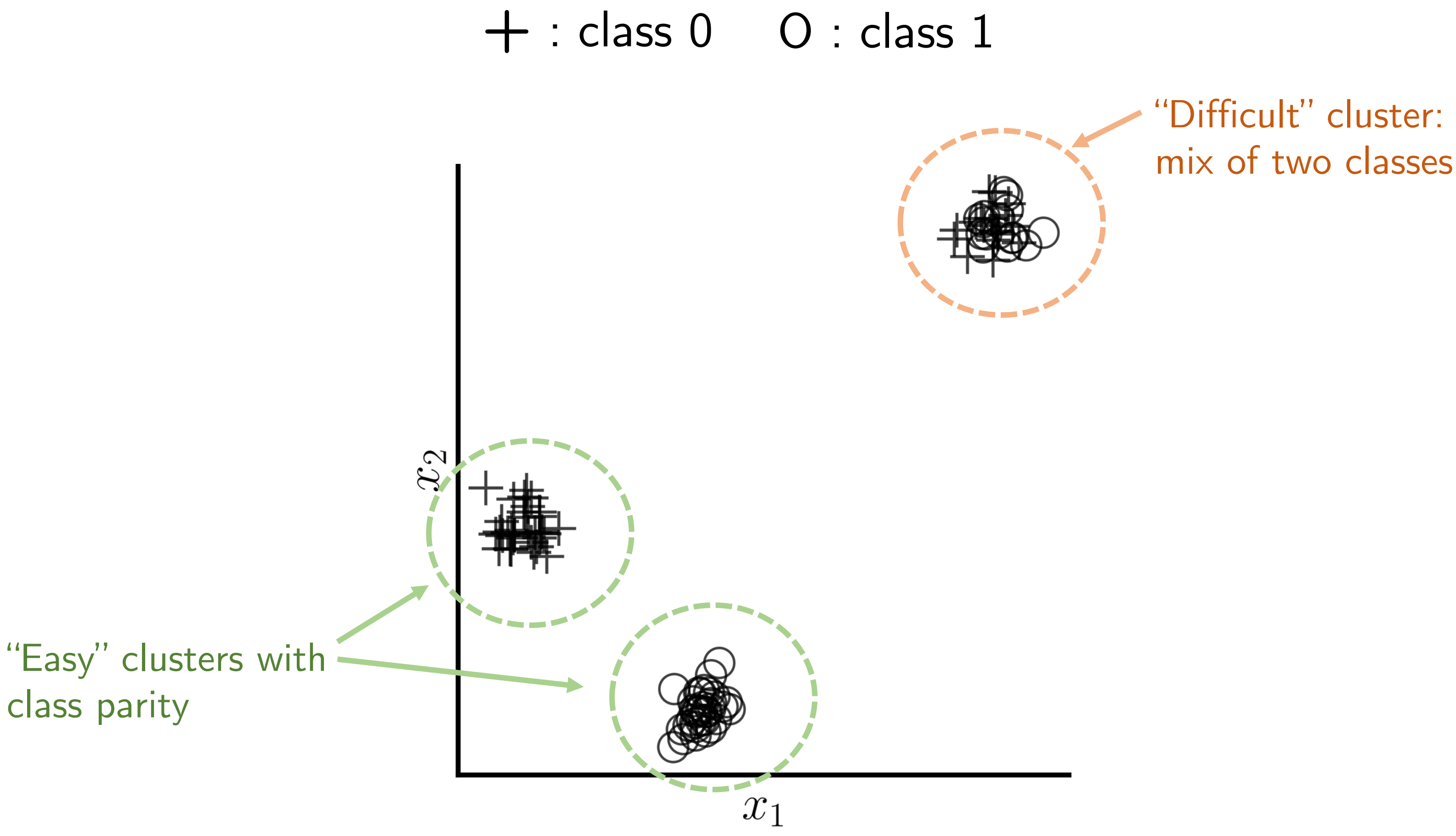
Experiment: Synthetic data



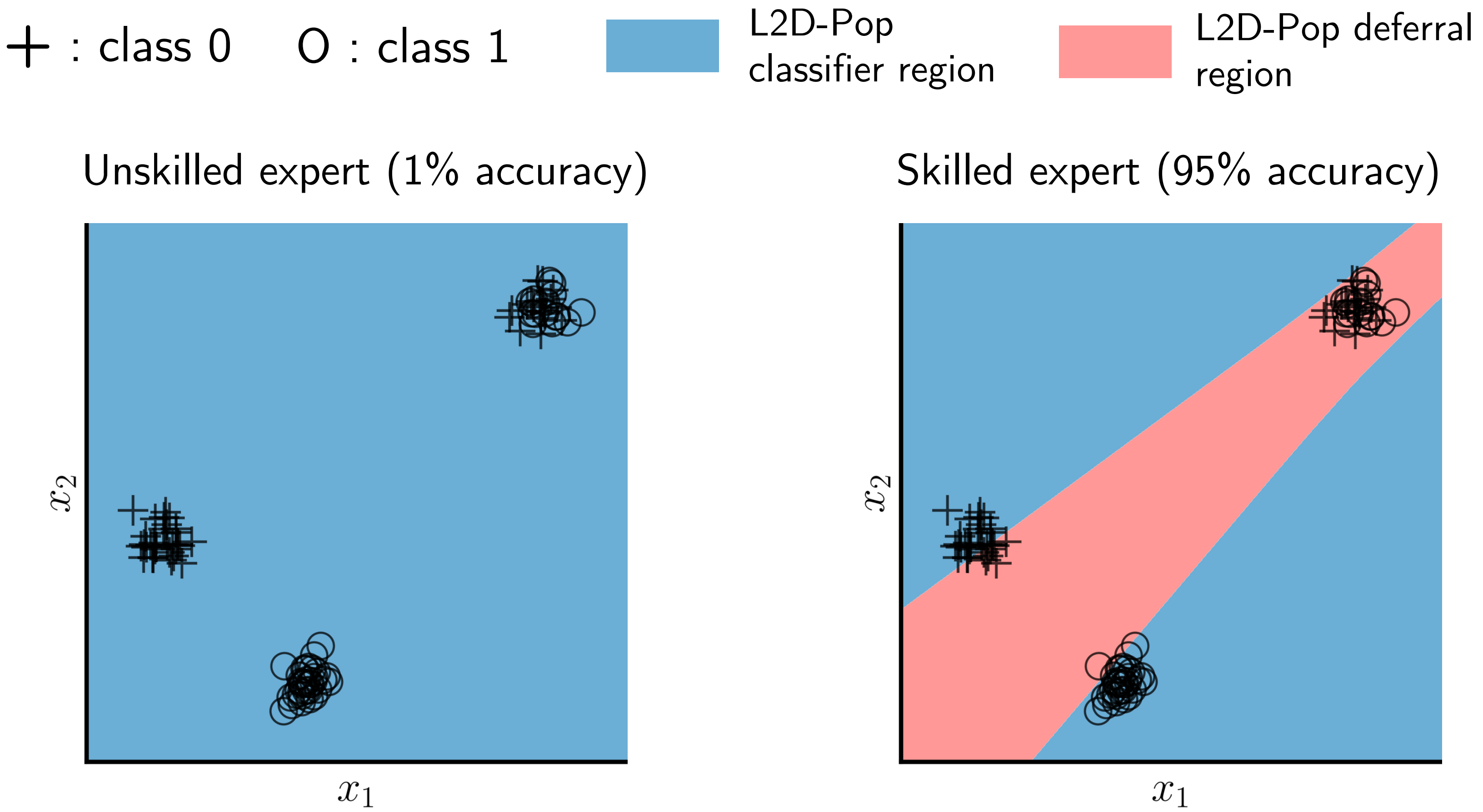
Experiment: Synthetic data



Experiment: Synthetic data

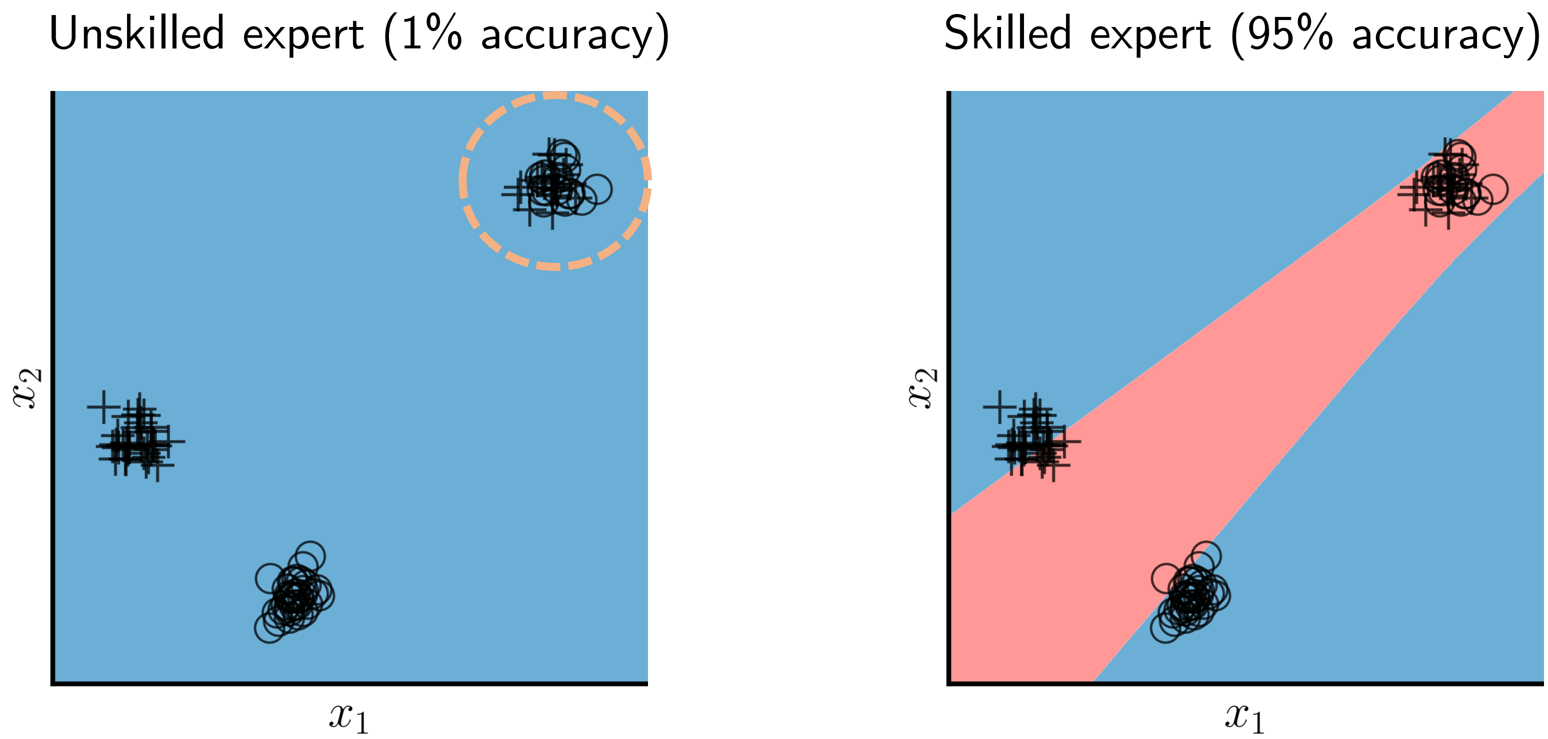


Experiment: Synthetic data



Experiment: Synthetic data

+ : class 0   O : class 1   L2D-Pop classifier region   L2D-Pop deferral region

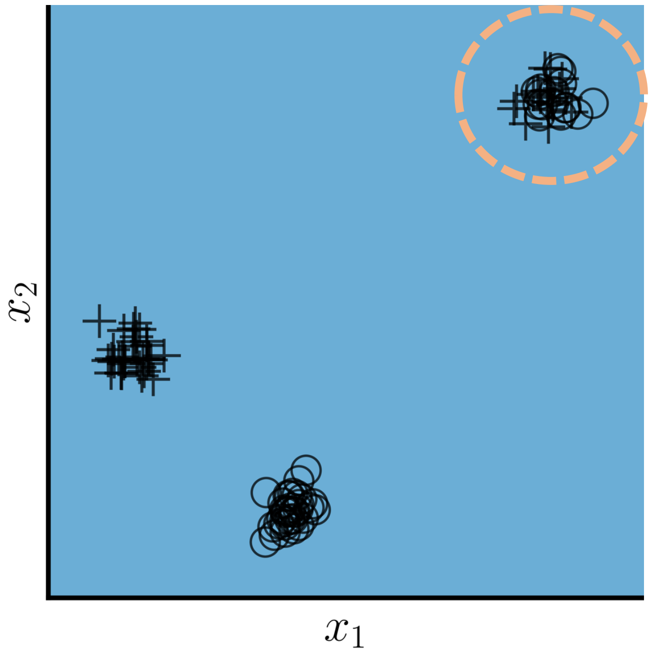


L2D-Pop (adaptive) ✓ Doesn't defer when the expert is poor

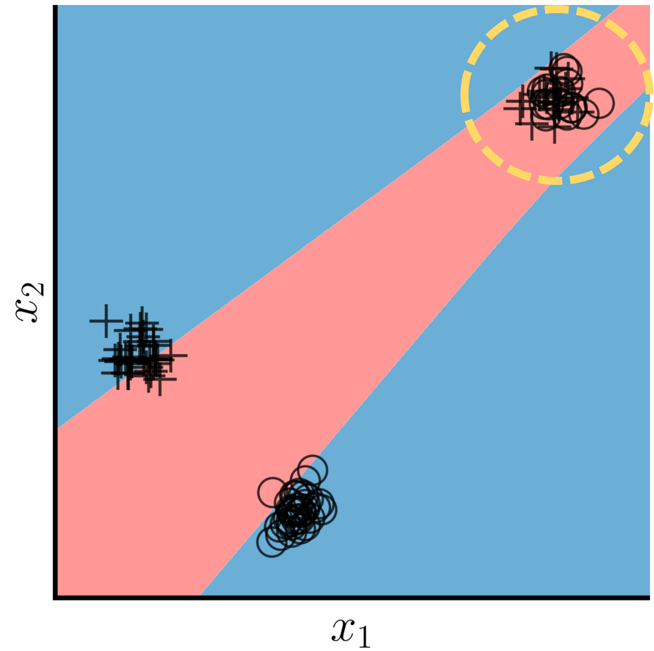
Experiment: Synthetic data

+ : class 0   O : class 1   L2D-Pop classifier region   L2D-Pop deferral region

Unskilled expert (1% accuracy)



Skilled expert (95% accuracy)



L2D-Pop (adaptive)

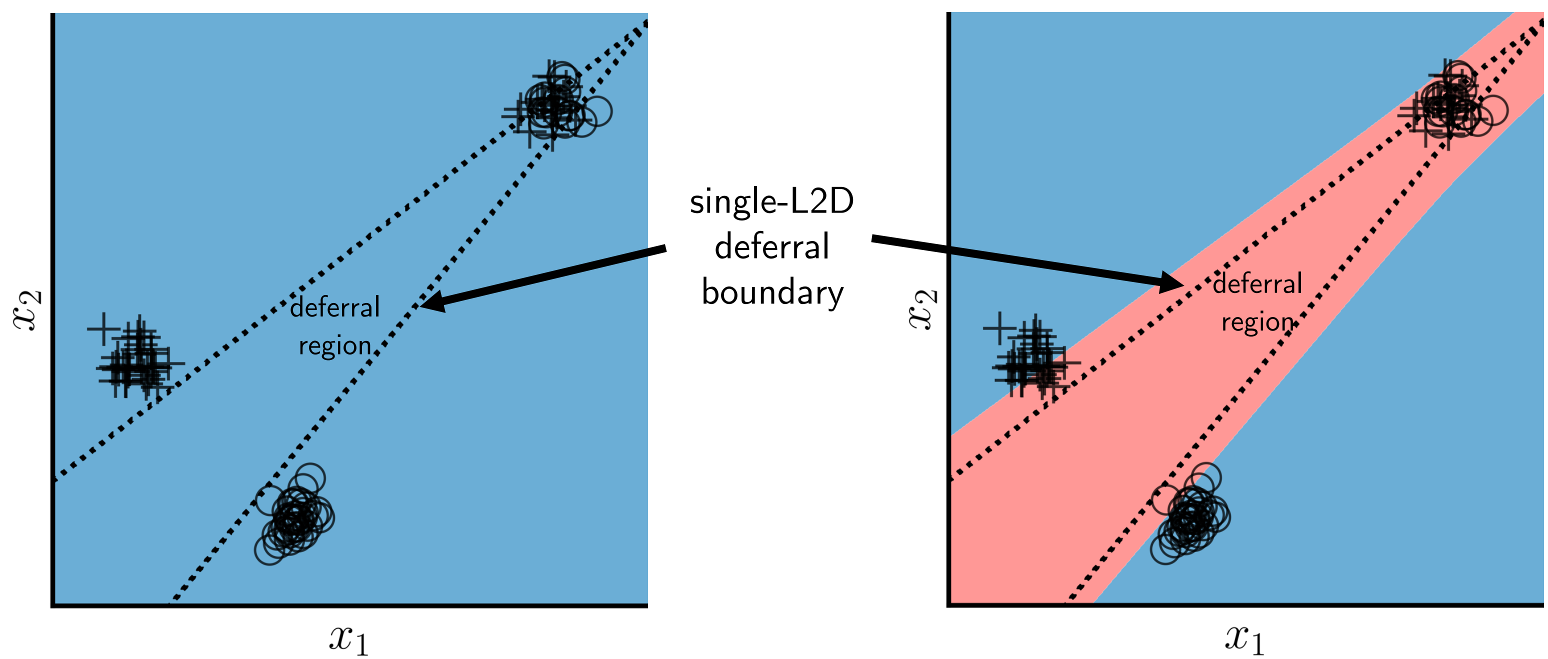
✓ Doesn't defer when the expert is poor

✓ Defers whole of difficult cluster when expert is good

Experiment: Synthetic data

+ : class 0   O : class 1   L2D-Pop classifier region   L2D-Pop deferral region

Unskilled expert (1% accuracy)   Skilled expert (95% accuracy)

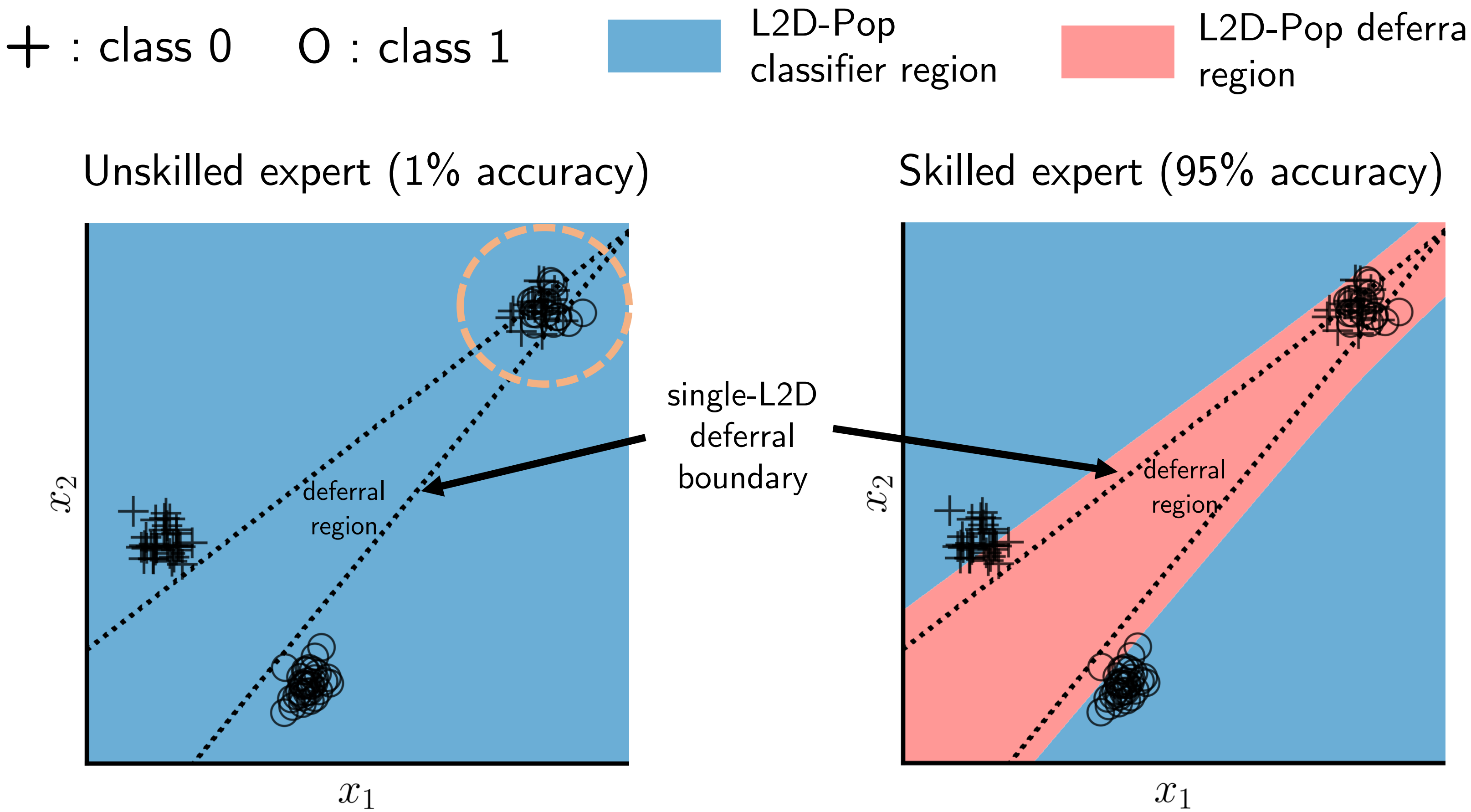


L2D-Pop (adaptive)

✓ Doesn't defer when the expert is poor

✓ Defers whole of difficult cluster when expert is good

Experiment: Synthetic data

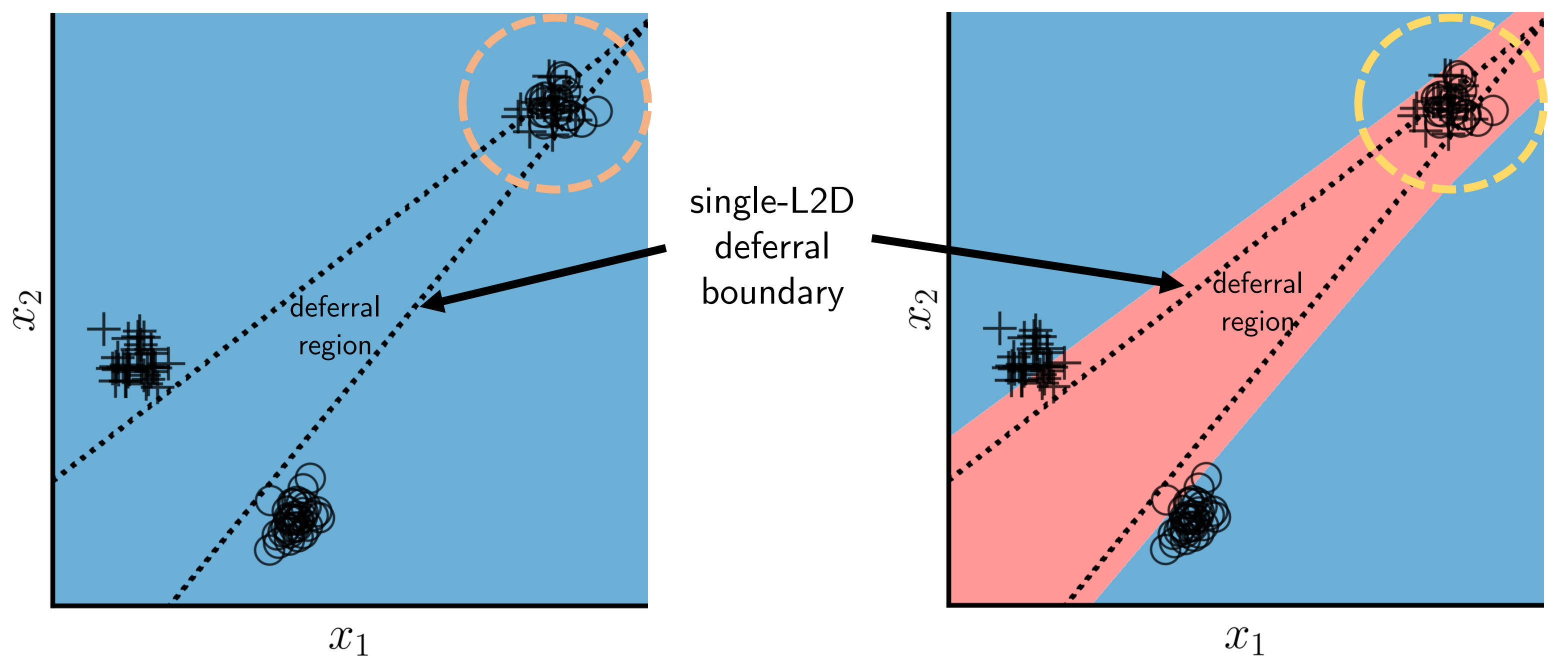


L2D-Pop (adaptive)	✓ Doesn't defer when the expert is poor	✓ Defers whole of difficult cluster when expert is good
single-L2D (constant)	X Over-defers as expert does worse than random on difficult cluster	

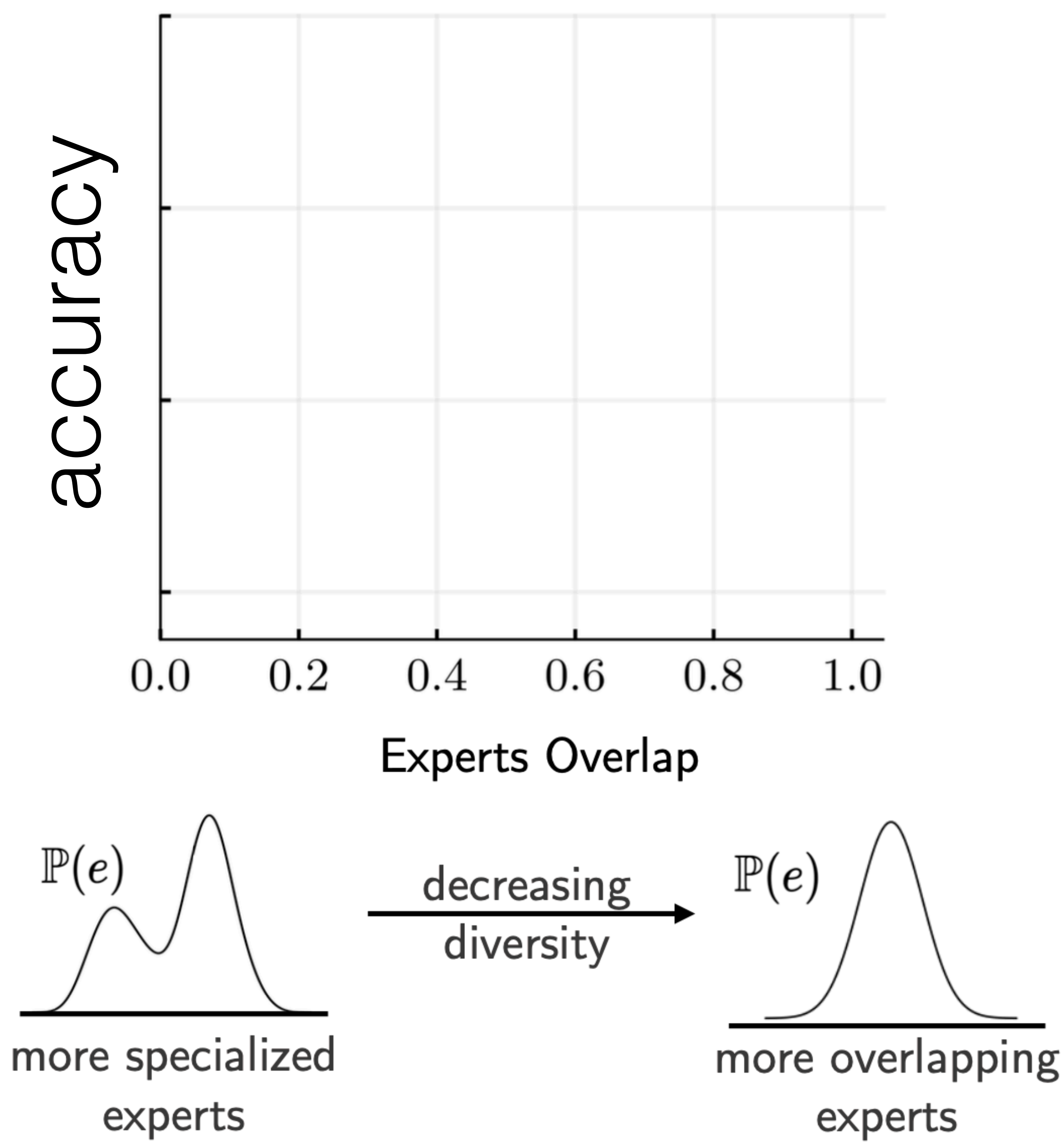
Experiment: Synthetic data

+ : class 0   O : class 1   L2D-Pop classifier region   L2D-Pop deferral region

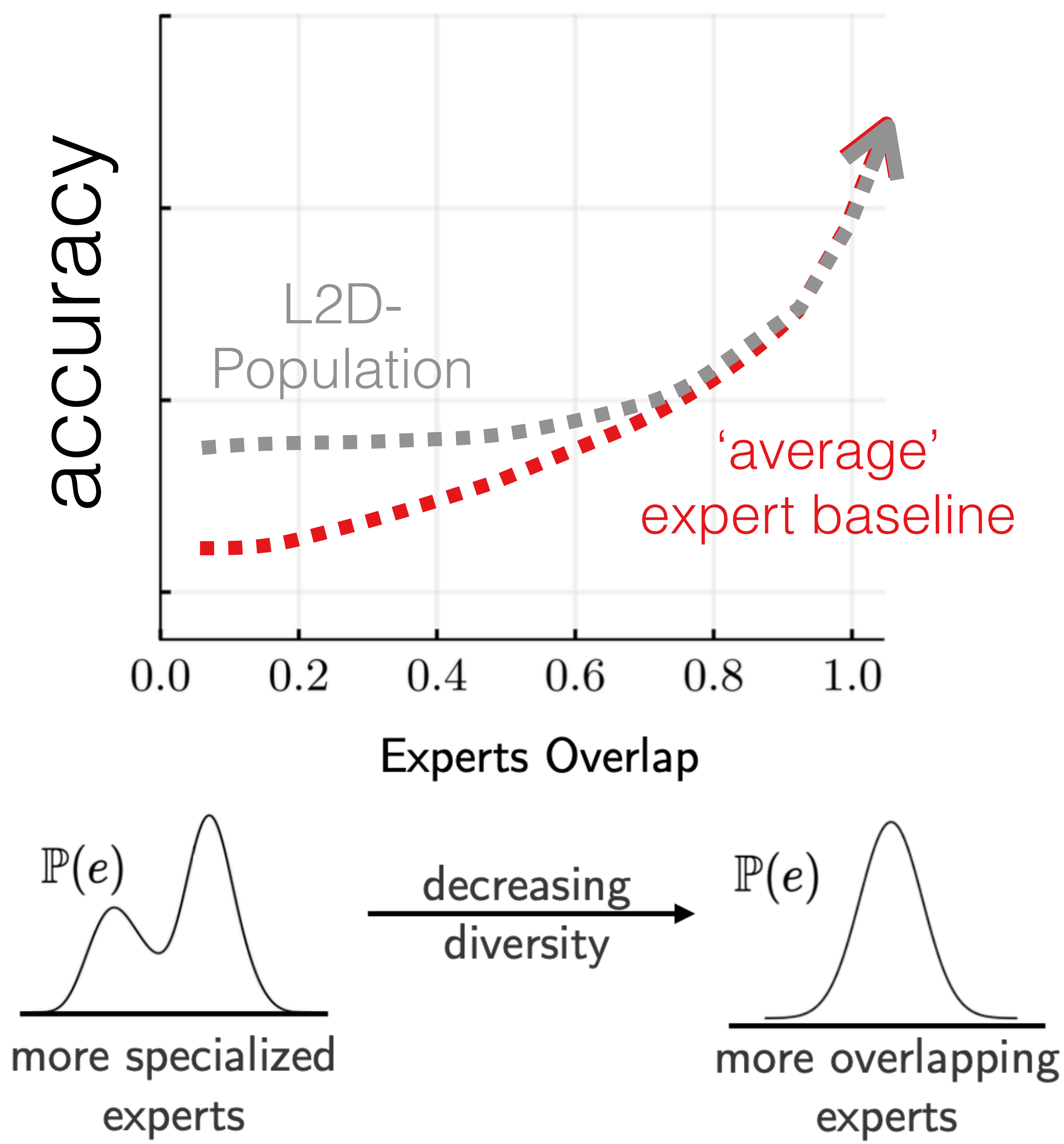
Unskilled expert (1% accuracy)   Skilled expert (95% accuracy)

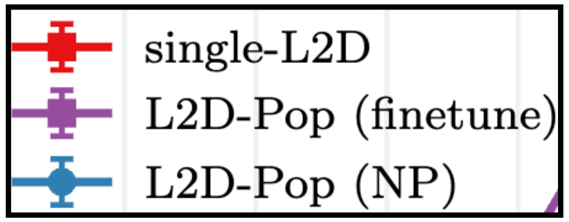


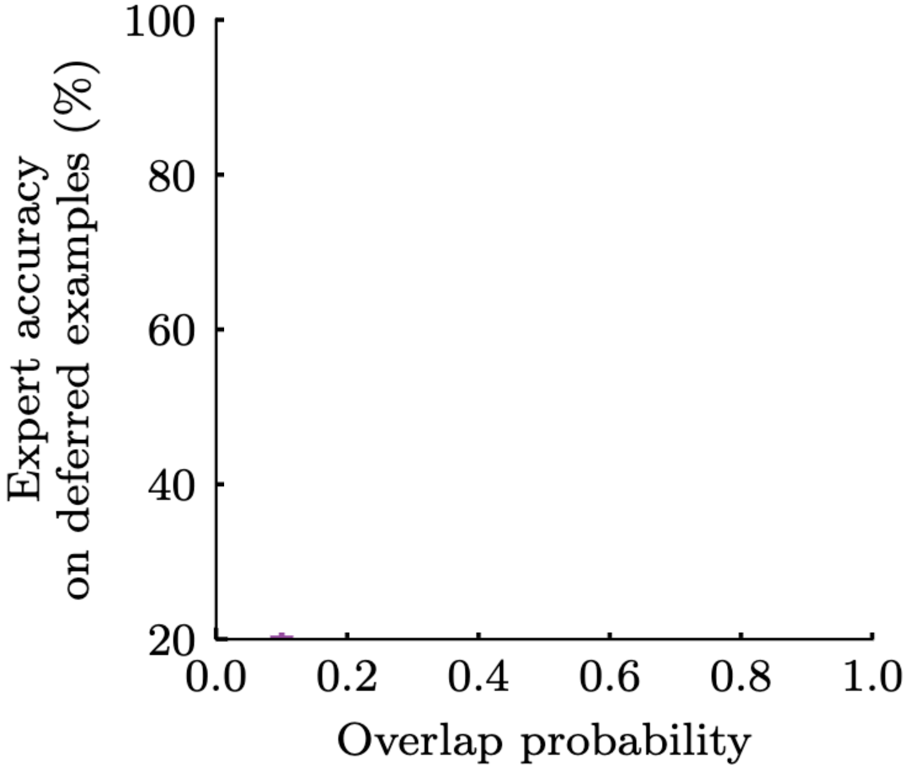
L2D-Pop (adaptive)	✓ Doesn't defer when the expert is poor	✓ Defers whole of difficult cluster when expert is good
single-L2D (constant)	X Over-defers as expert does worse than random on difficult cluster	X Under-defers as classifier only has random chance of being correct on difficult cluster



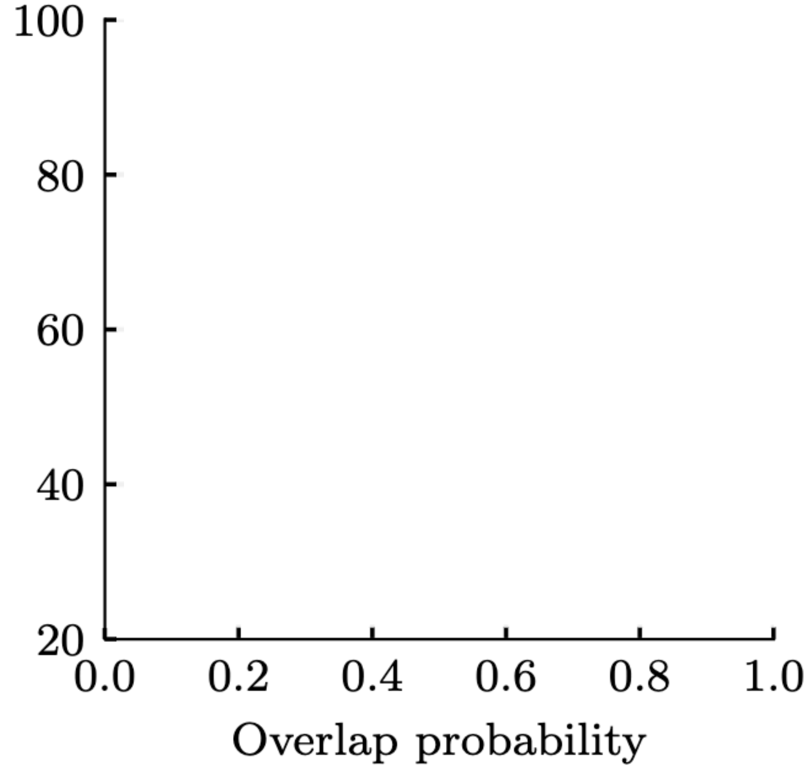
accuracy
0.0
0.2
0.4
0.6
0.8
1.0
Experts Overlap
$\mathbb{P}(e)$
decreasing
diversity
$\mathbb{P}(e)$
more specialized
experts
more overlapping
experts


accuracy
L2D-
Population
'average'
expert baseline
0.0
0.2
0.4
0.6
0.8
1.0
Experts Overlap
$\mathbb{P}(e)$
decreasing
diversity
$\mathbb{P}(e)$
more specialized
experts
more overlapping
experts

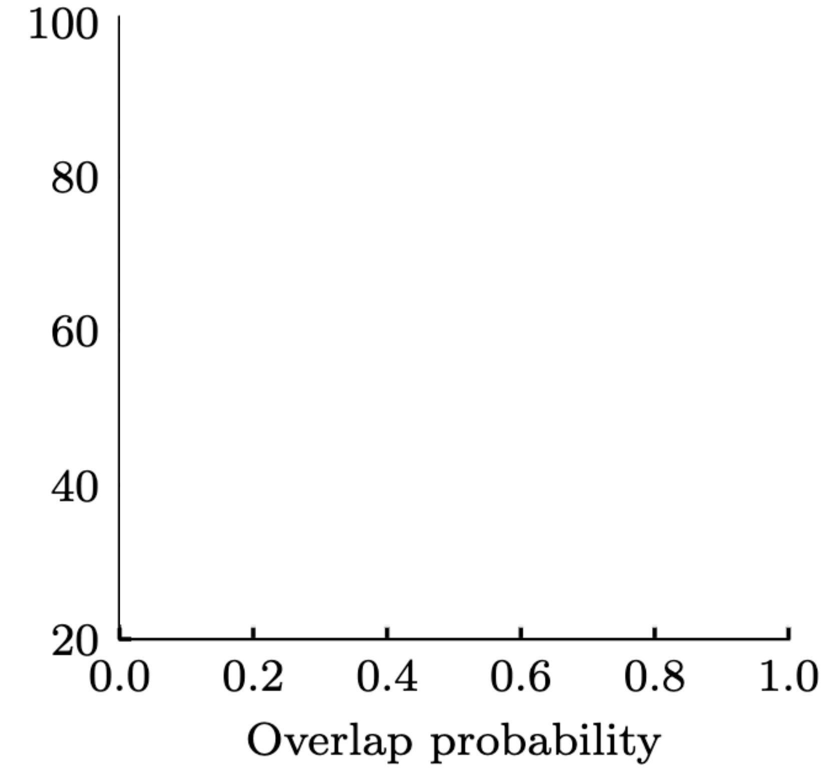




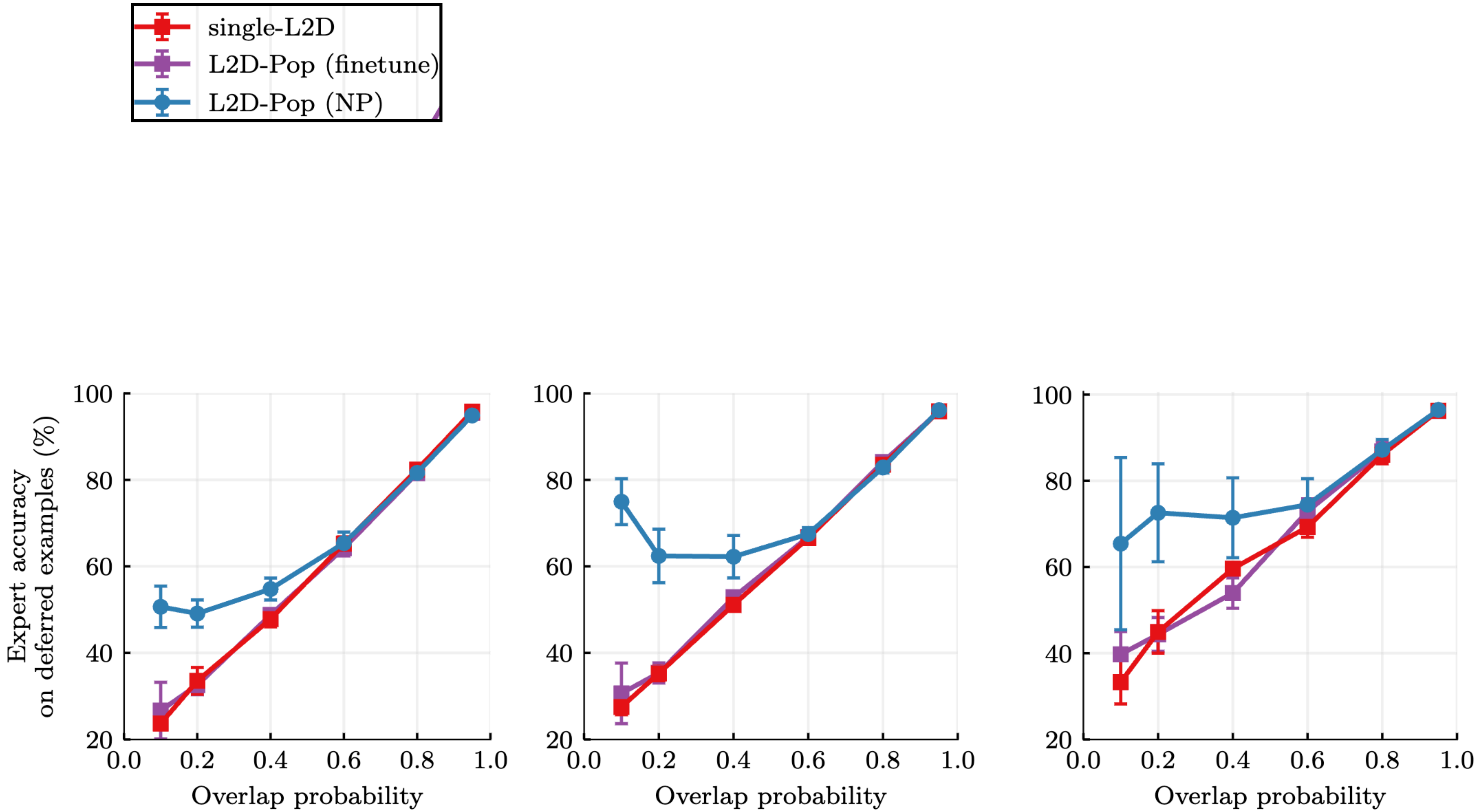
(a) Traffic Signs



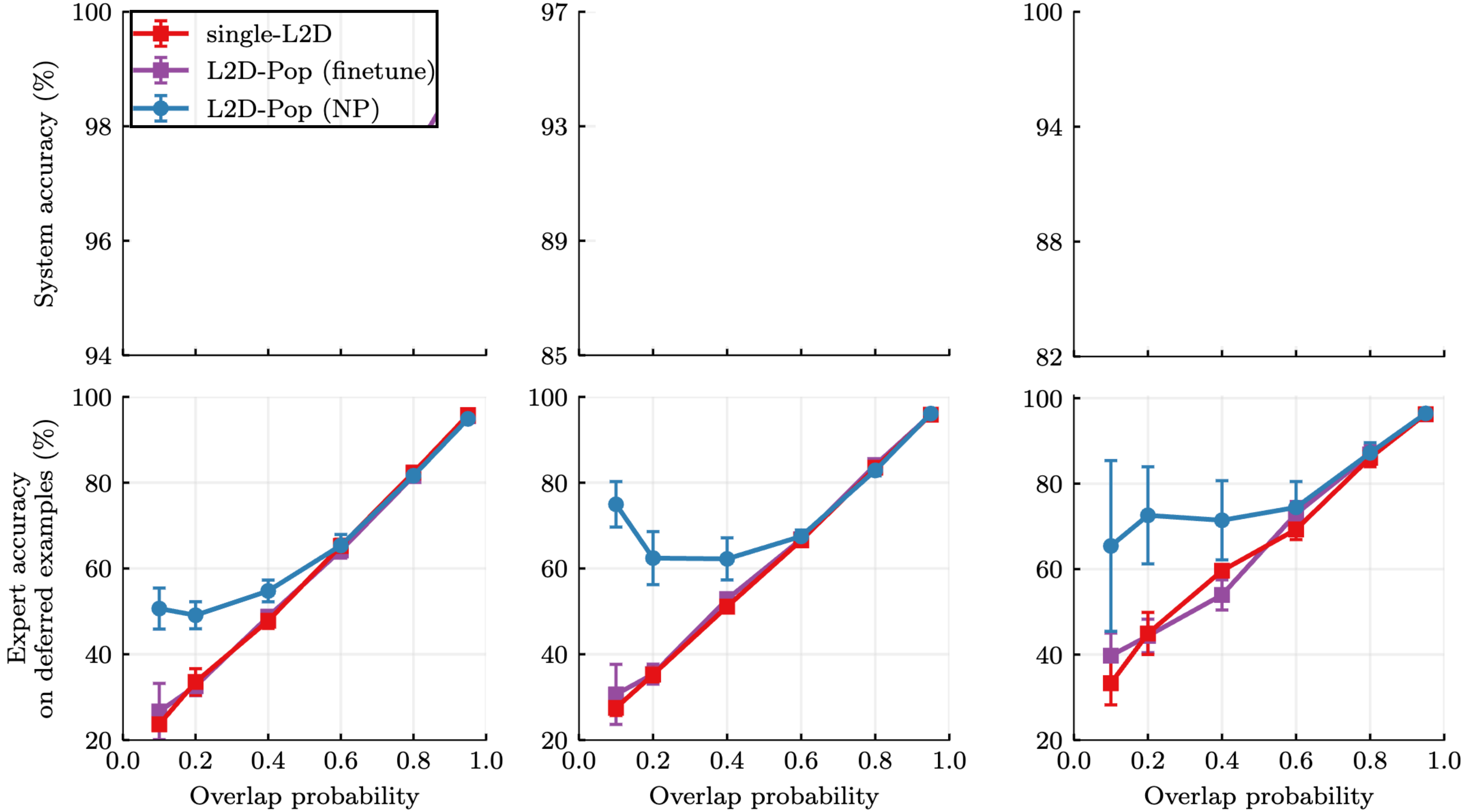
(b) CIFAR-10



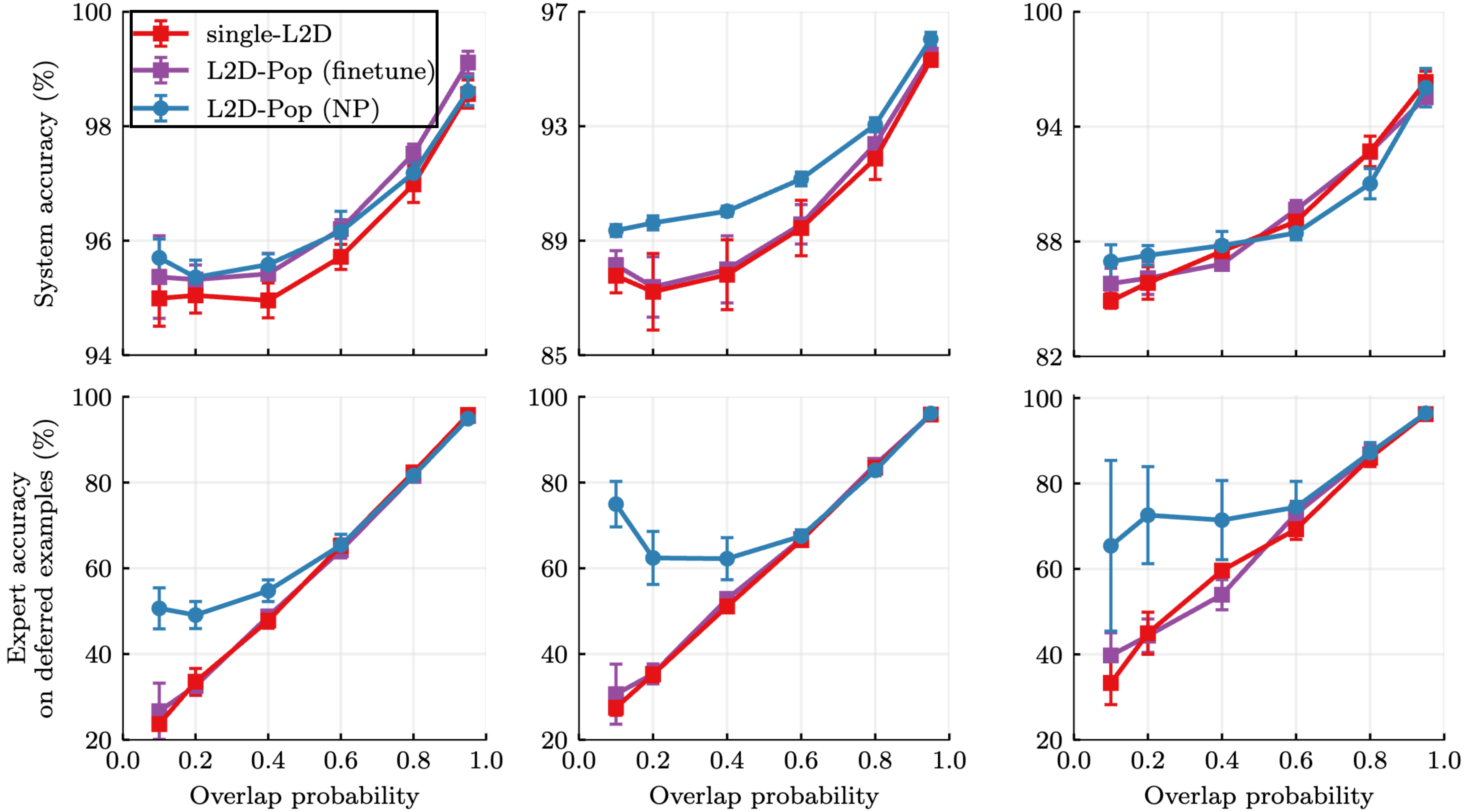
(c) HAM10000



(a) Traffic Signs (b) CIFAR-10 (c) HAM10000



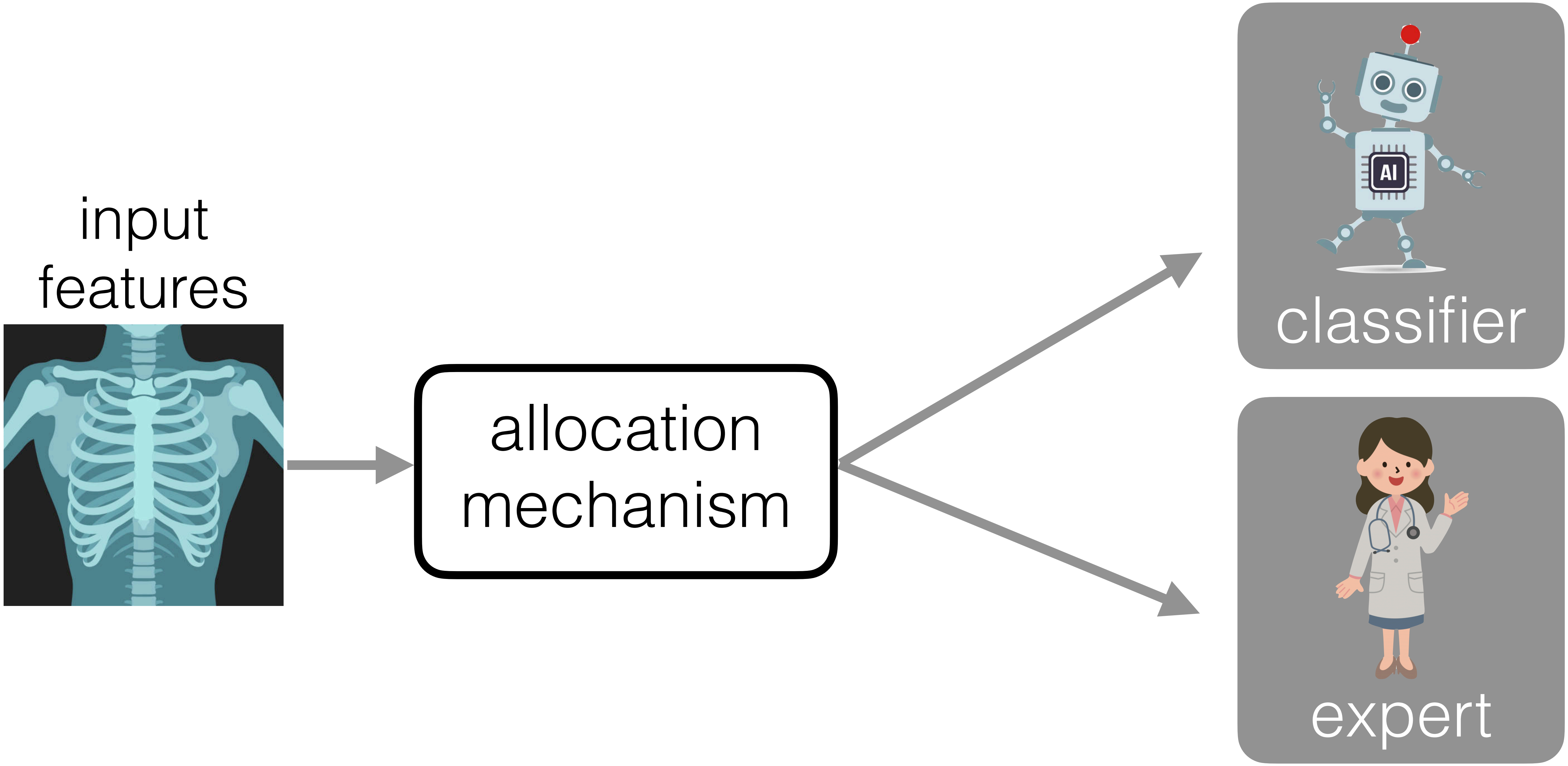
(a) Traffic Signs (b) CIFAR-10 (c) HAM10000

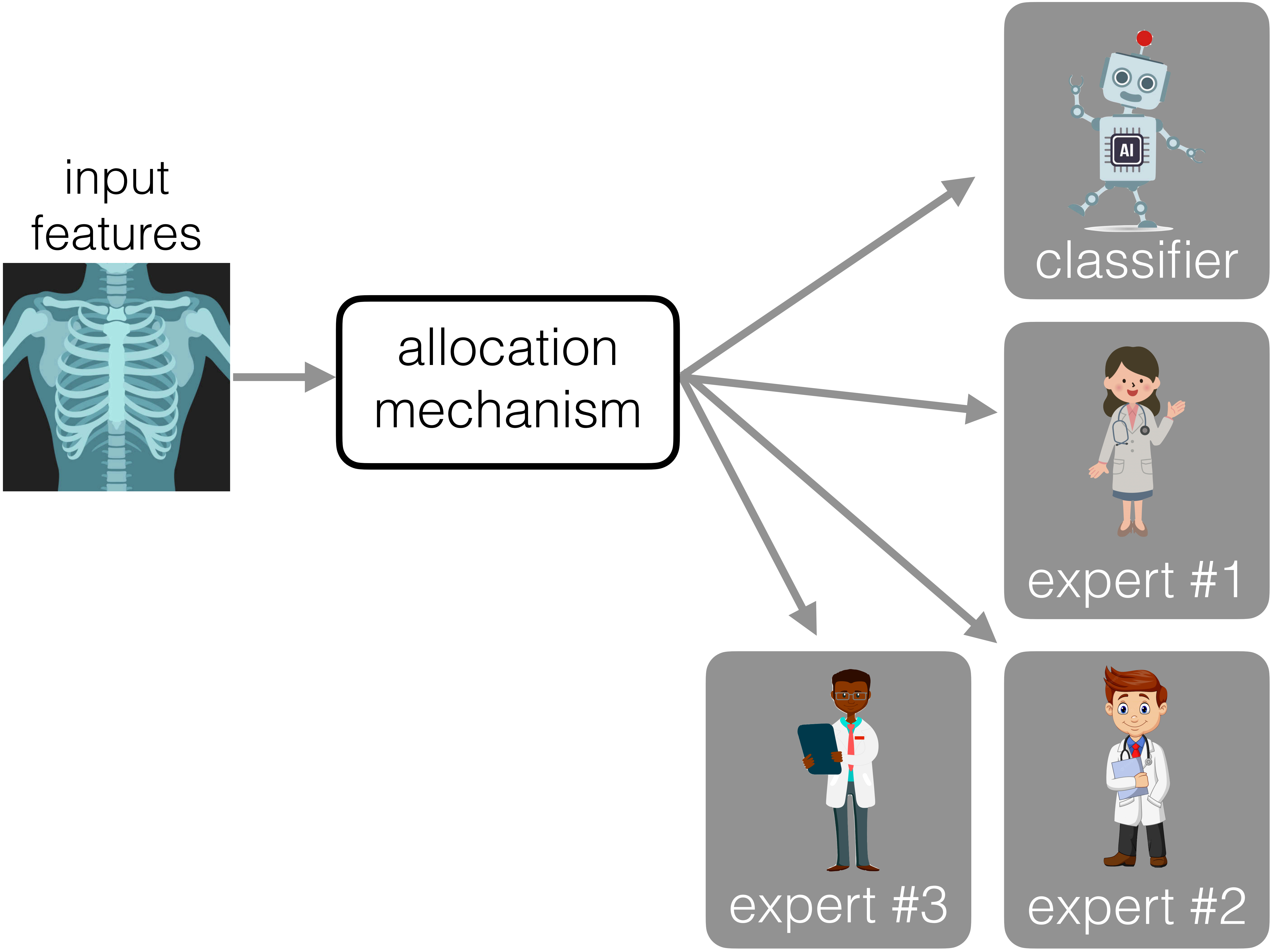


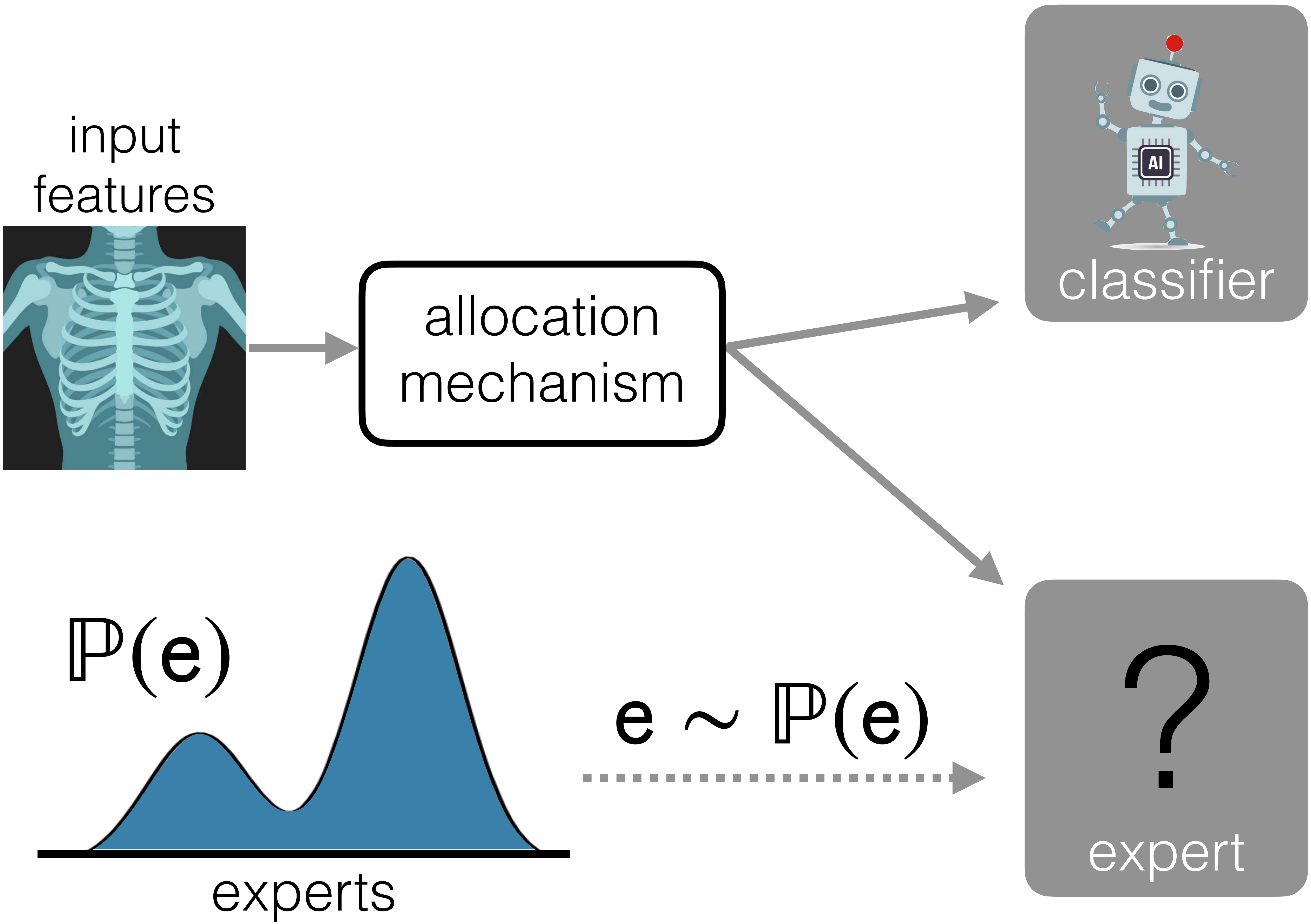
(a) Traffic Signs (b) CIFAR-10 (c) HAM10000

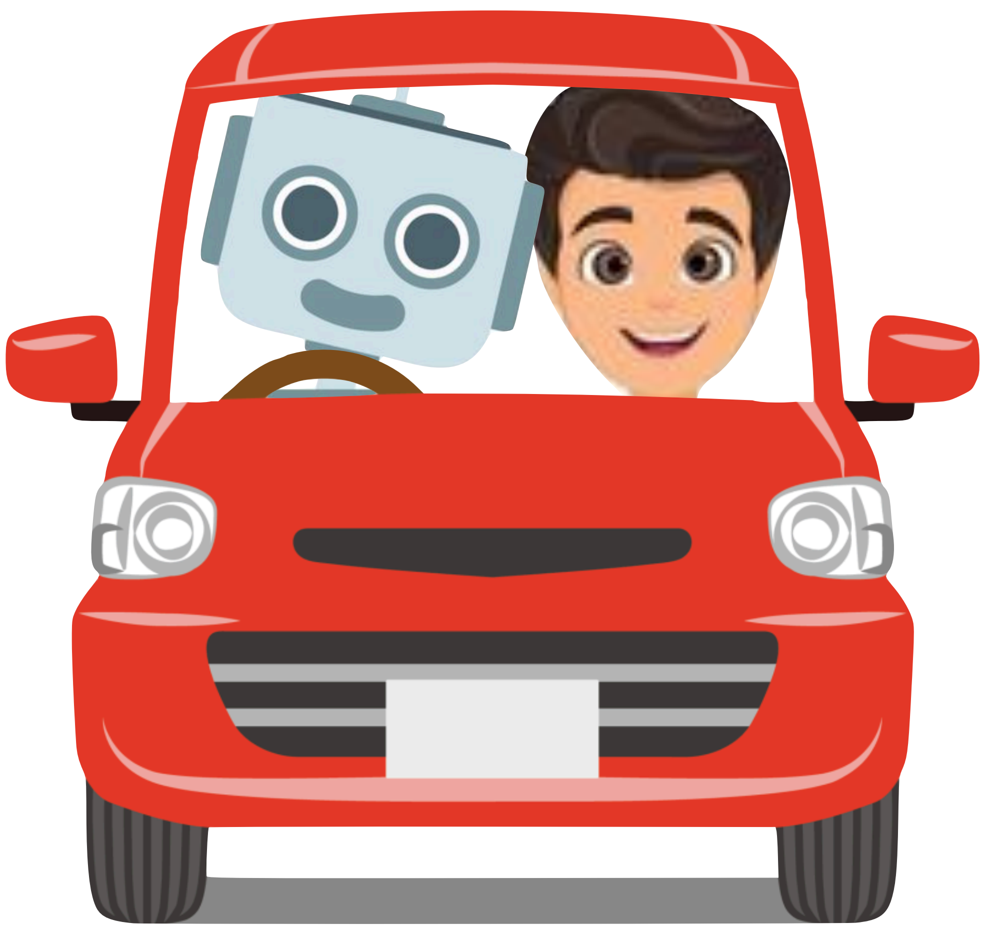
- ⊗ single expert
  - ⊗ softmax surrogate loss
  - ⊗ improving calibration via one-vs-all
- ⊗ multiple experts
  - ⊗ surrogate losses
  - ⊗ conformal sets of experts
- ⊗ population of experts
  - ⊗ surrogate losses
  - ⊗ meta-learning a rejector

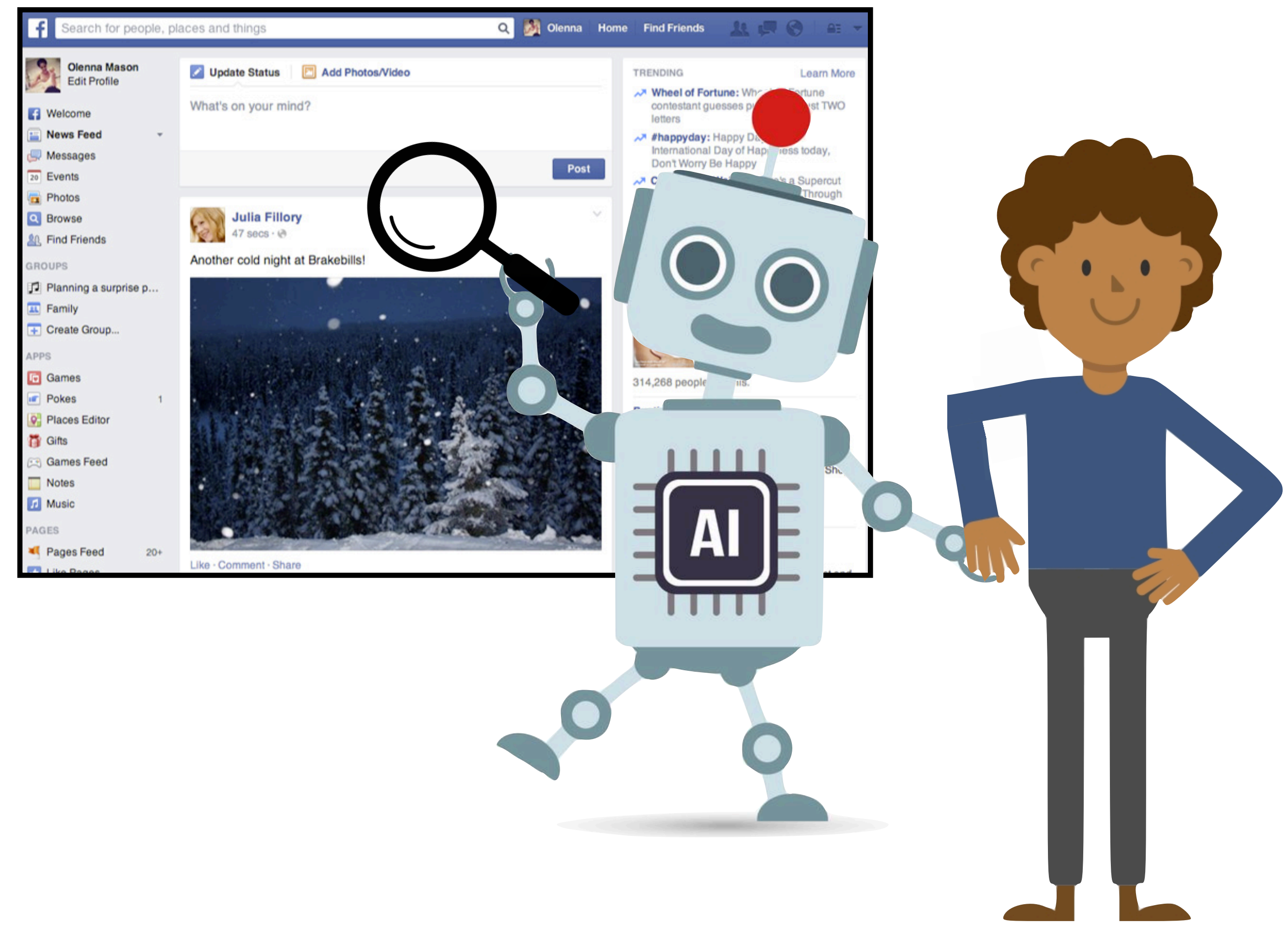
- ⊗ single expert
  - ⊗ softmax surrogate loss
  - ⊗ improving calibration via one-vs-all
- ⊗ multiple experts
  - ⊗ surrogate losses
  - ⊗ conformal sets of experts
- ⊗ population of experts
  - ⊗ surrogate losses
  - ⊗ meta-learning a rejector

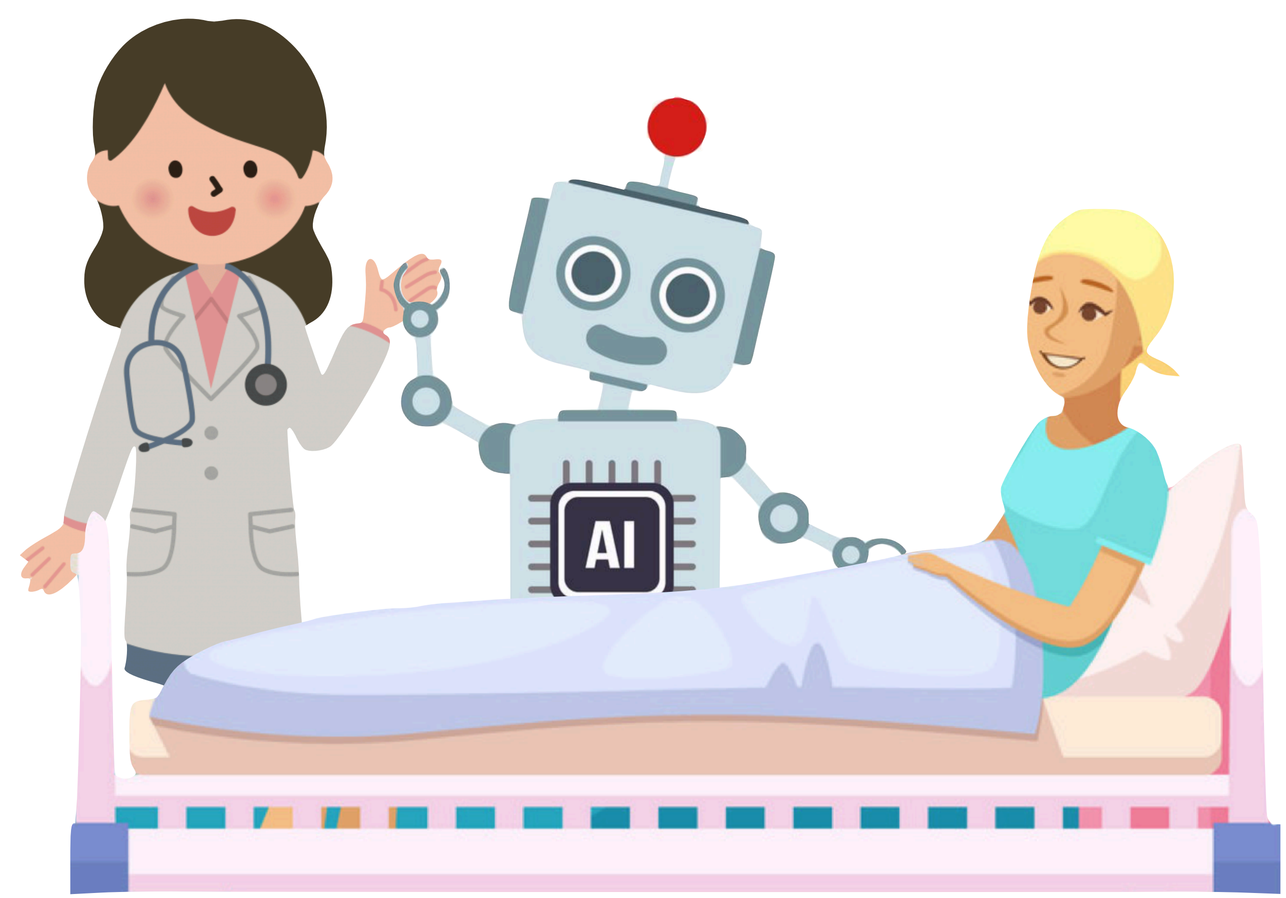

input
features
allocation
mechanism
AI
classifier
expert


input
features
allocation
mechanism
classifier
expert #1
expert #2
expert #3


input
features
allocation
mechanism
classifier
AI
$\mathbb{P}(\mathbf{e})$
$\mathbf{e} \sim \mathbb{P}(\mathbf{e})$
?
expert
experts




Search for people, places and things
Olenna
Home
Find Friends
Olenna Mason
Edit Profile
Welcome
News Feed
Messages
Events
Photos
Browse
Find Friends
GROUPS
Planning a surprise p...
Family
Create Group...
APPS
Games
Pokes
Places Editor
Gifts
Games Feed
Notes
Music
PAGES
Pages Feed
Update Status
Add Photos/Video
What's on your mind?
Post
Julia Fillory
47 secs
Another cold night at Brakebills!
Like · Comment · Share
TRENDING
Learn More
314,268 people
AI


AI

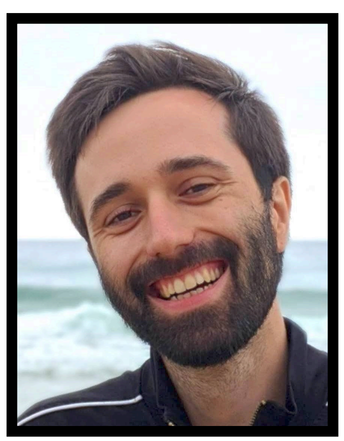
papers & code

funding provided by

NWO
Netherlands Organisation for Scientific Research

co-authors


Rajeev Verma


Daniel Barrejón


Dharmesh Tailor


Putra Manggala


Aditya Patra

Appendix

0-1 loss

$$\ell(\mathrm{r}, \mathrm{h}; \mathfrak{D}) =$$

$$\sum_n (1 - \mathrm{r}(\mathrm{x}_n))\ \mathbb{I}[\mathrm{h}(\mathrm{x}_n) \neq \mathrm{y}_n]\ +\ \mathrm{r}(\mathrm{x}_n)\ \mathbb{I}[\mathrm{m}_n \neq \mathrm{y}_n]$$

classifier loss — expert loss

estimators

single expert

softmax: $\hat{\mathsf{p}}(\mathsf{m} = \mathsf{y} \mid \mathsf{x}) = \dfrac{\mathsf{h}_\perp(\mathsf{x})}{1 - \mathsf{h}_\perp(\mathsf{x})}$

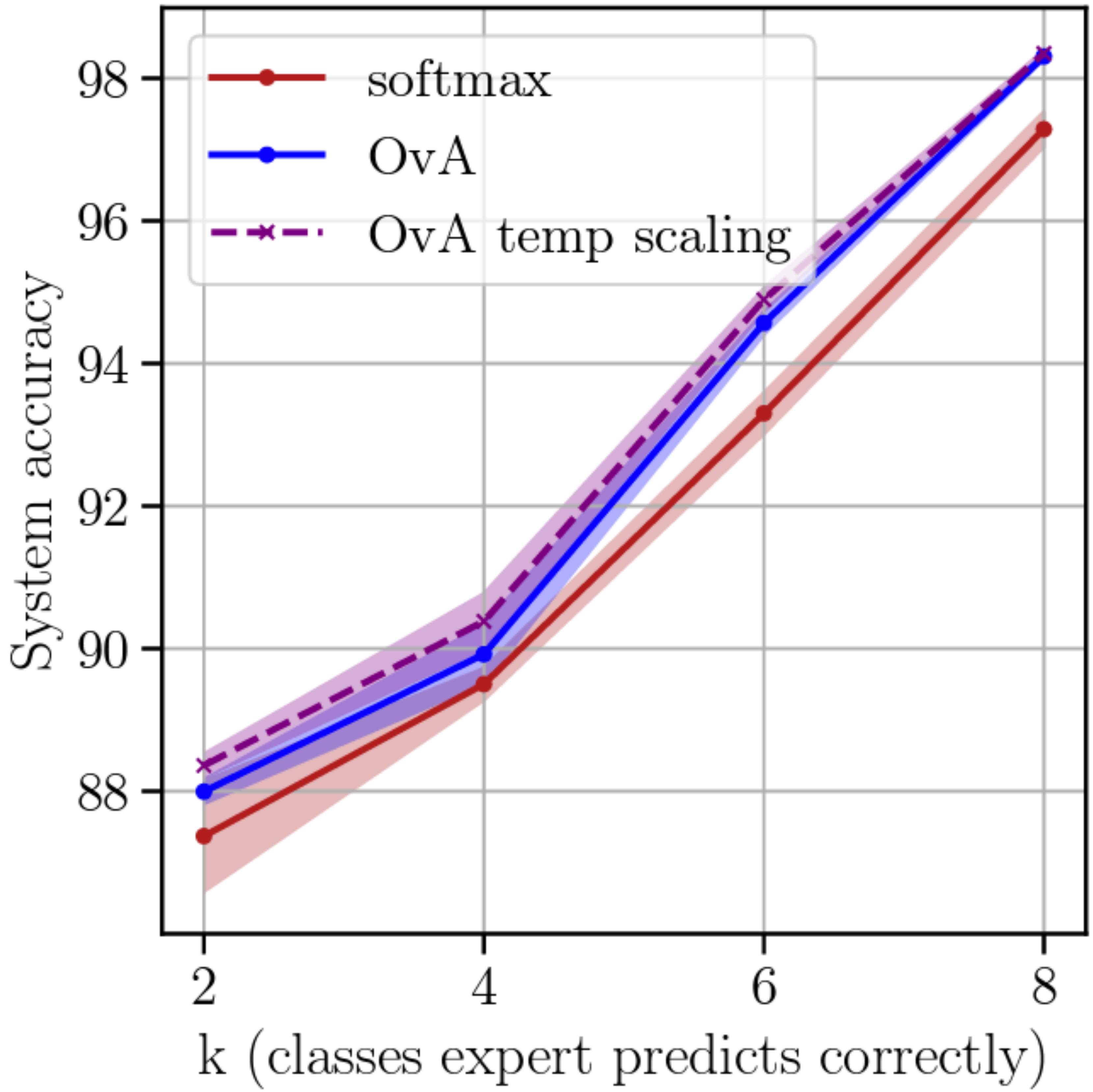
one-vs-all: $\hat{\mathsf{p}}(\mathsf{m} = \mathsf{y} \mid \mathsf{x}) = \mathsf{h}_\perp(\mathsf{x})$

---

multi-expert

softmax: $\hat{\mathsf{p}}(\mathsf{m}_j = \mathsf{y} \mid \mathsf{x}) = \dfrac{\mathsf{h}_{\perp,j}(\mathsf{x})}{1 - \sum_{e=1}^{J} \mathsf{h}_{\perp,e}(\mathsf{x})}$

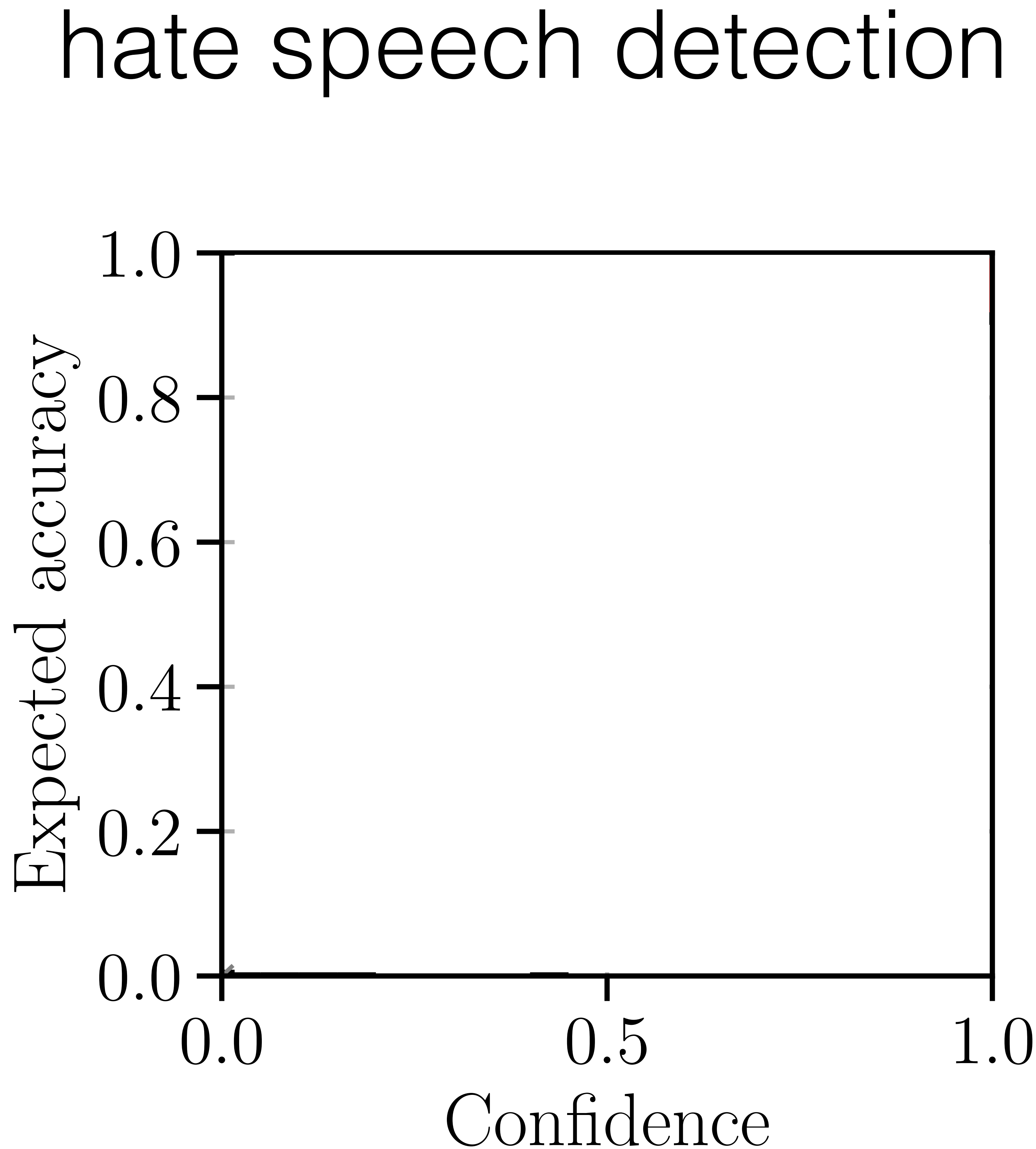
one-vs-all: $\hat{\mathsf{p}}(\mathsf{m}_j = \mathsf{y} \mid \mathsf{x}) = \mathsf{h}_{\perp,j}(\mathsf{x})$



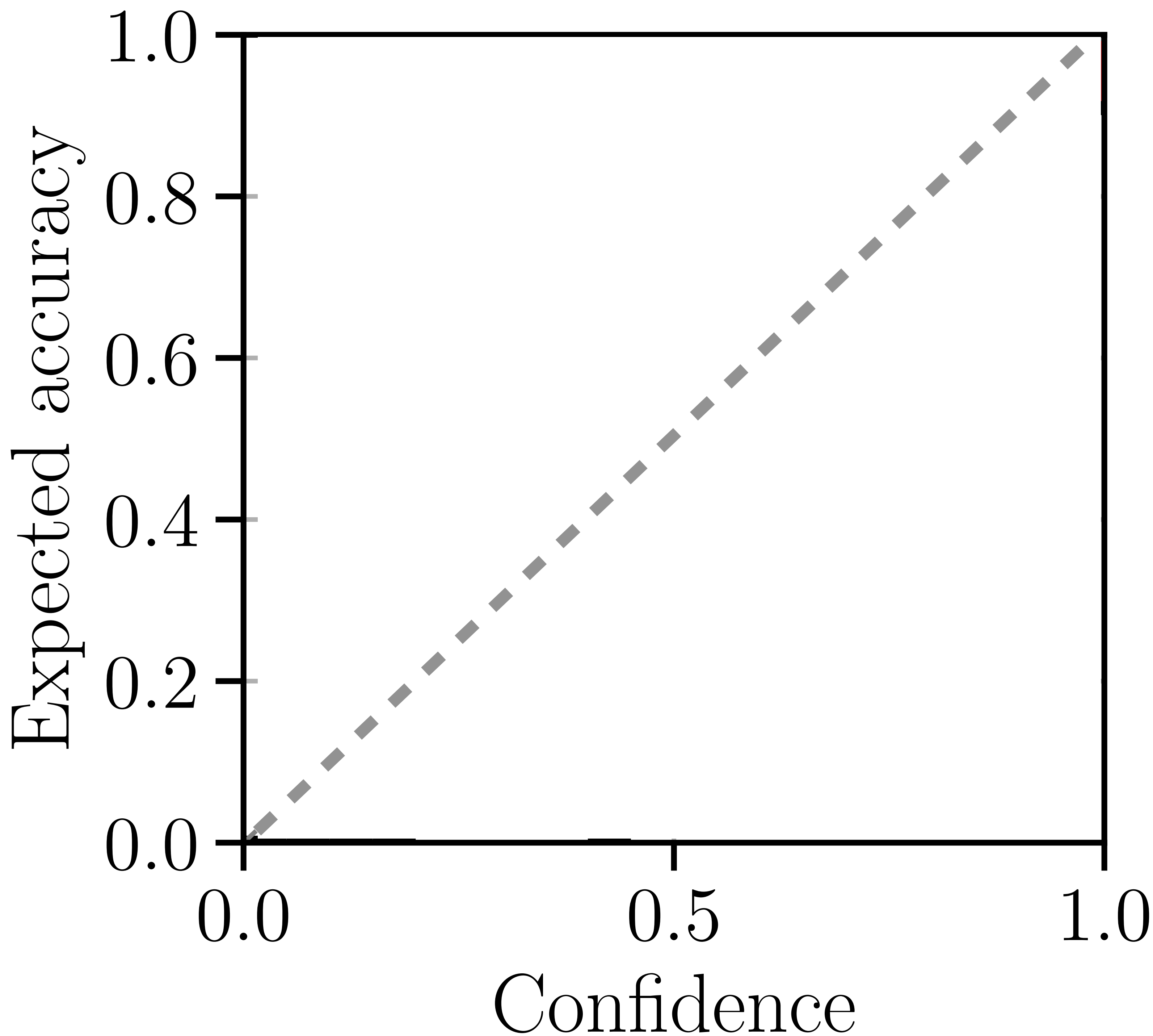
softmax
OvA
OvA temp scaling
System accuracy
98
96
94
92
90
88
2
4
6
8
k (classes expert predicts correctly)

hate speech detection

[Davidson et al., ICWSM 2017]


hate speech detection
1.0
0.8
0.6
0.4
0.2
0.0
Expected accuracy
0.0
0.5
1.0
Confidence

hate speech detection

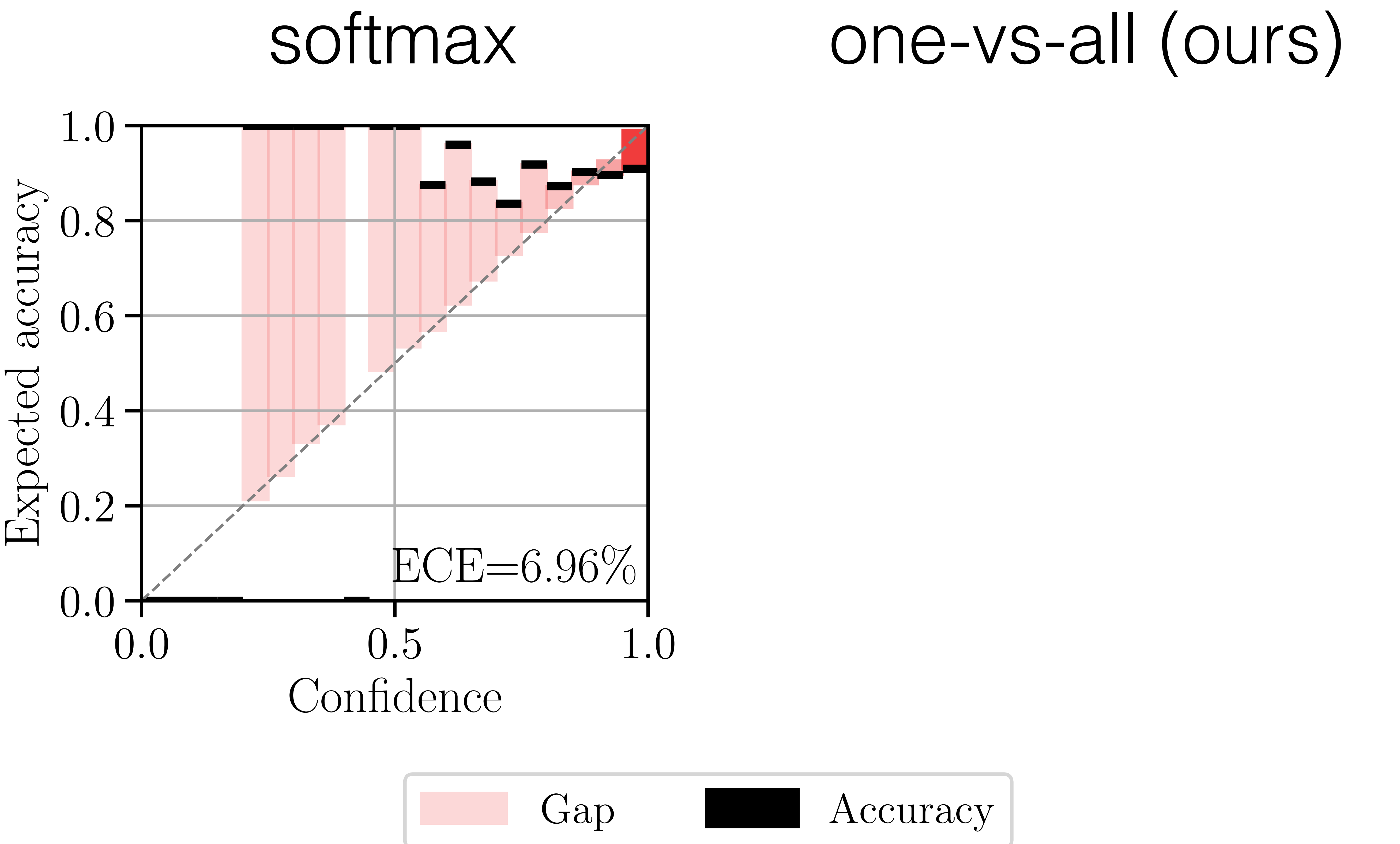


hate speech detection

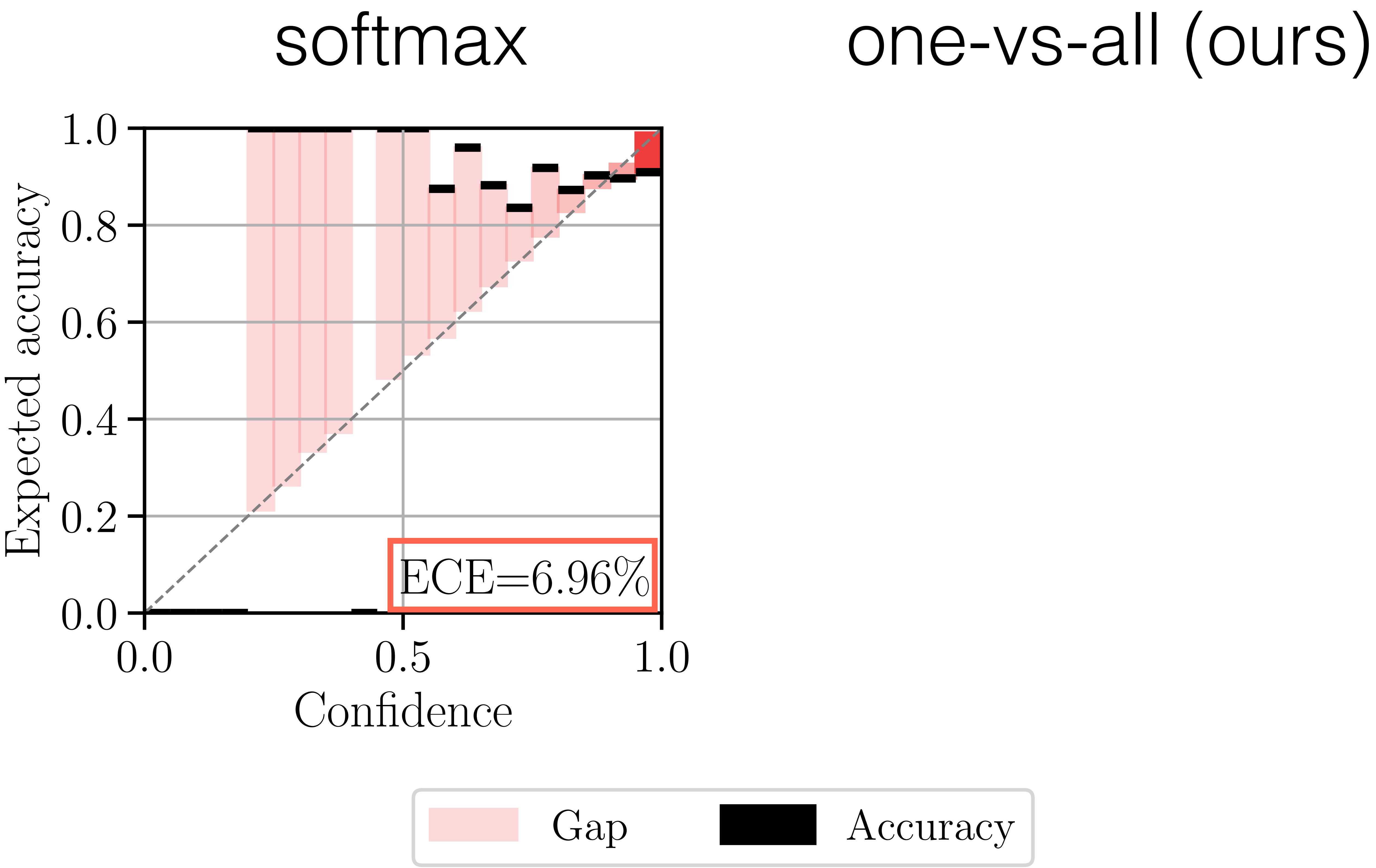
softmax

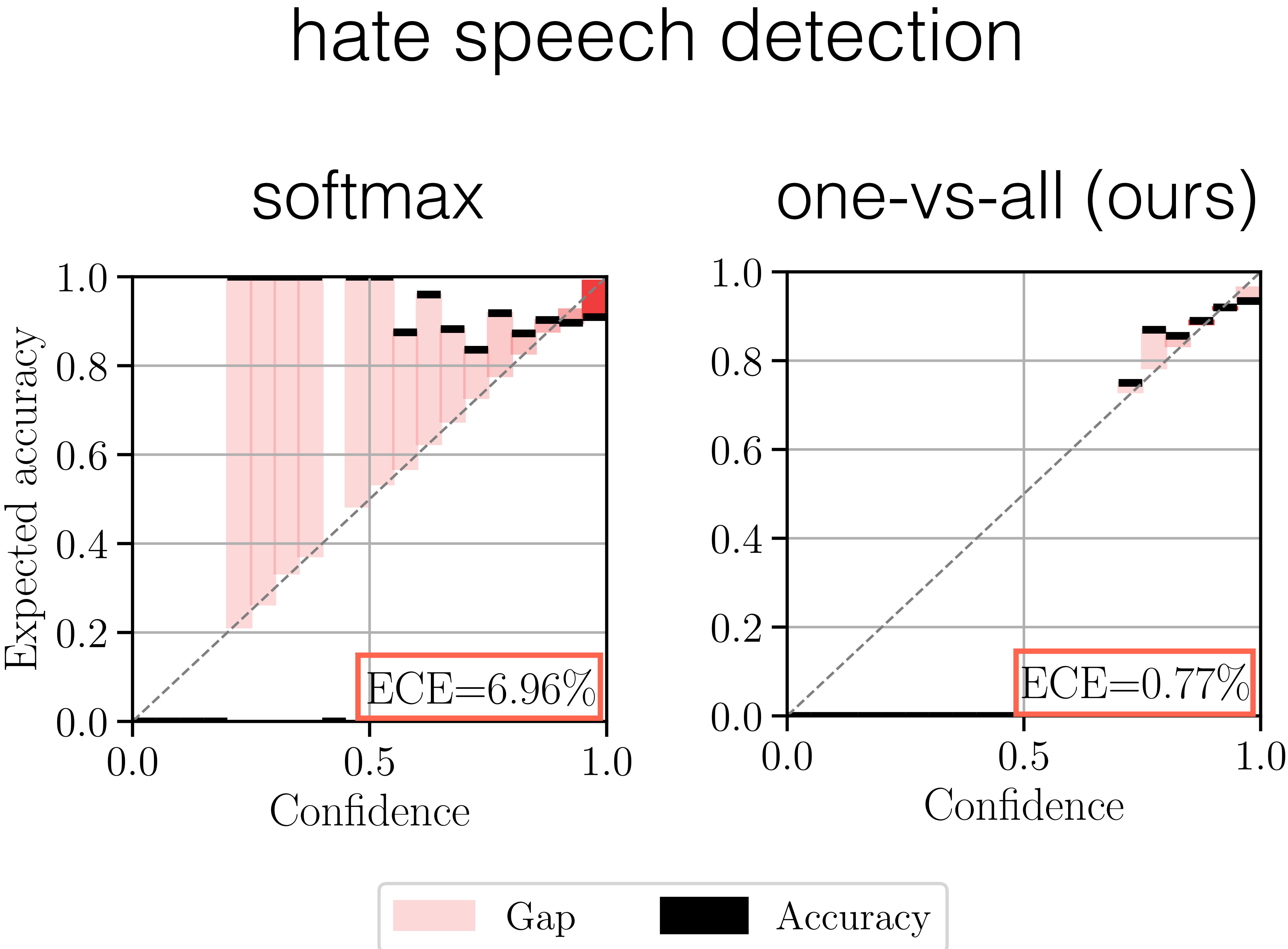
one-vs-all (ours)

hate speech detection

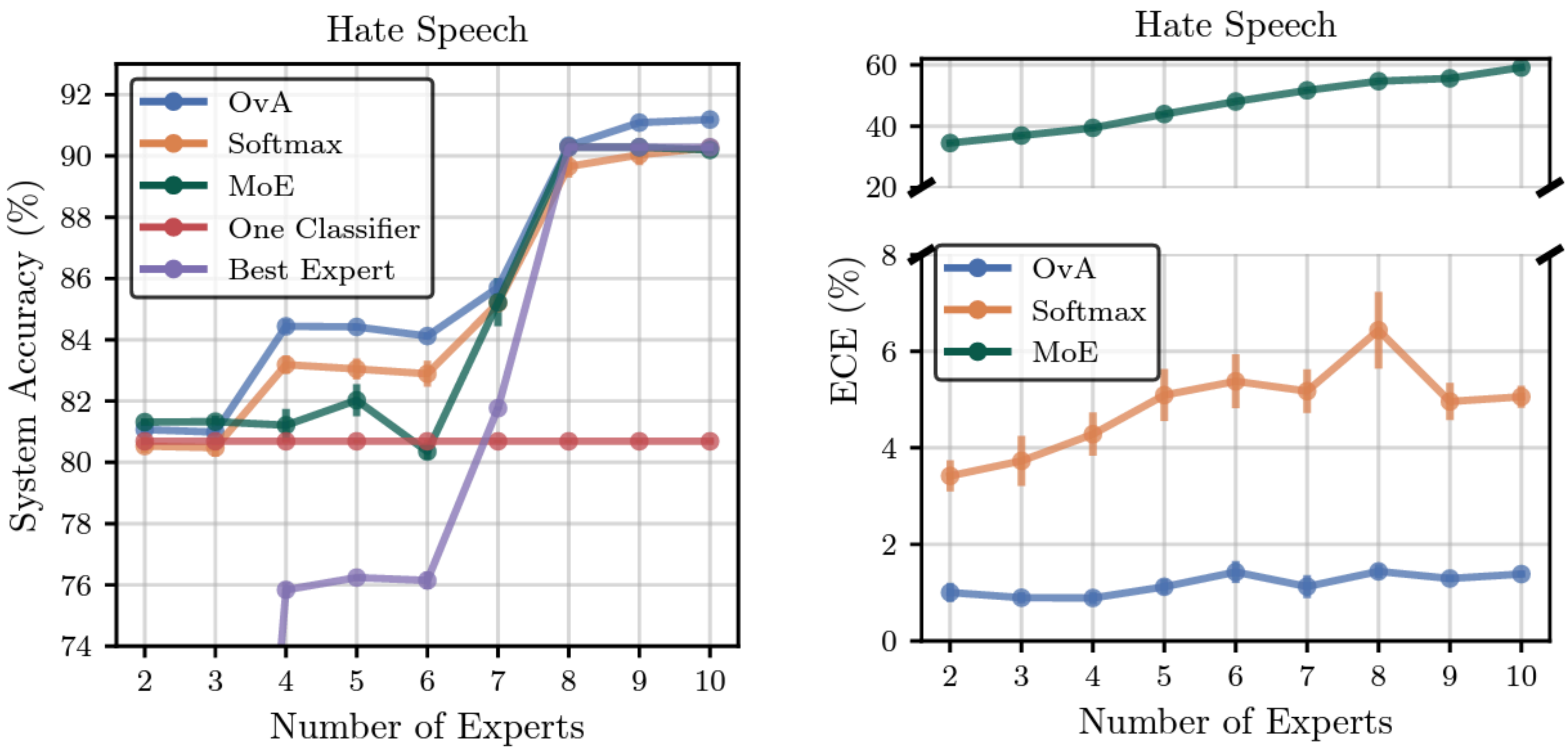


hate speech detection





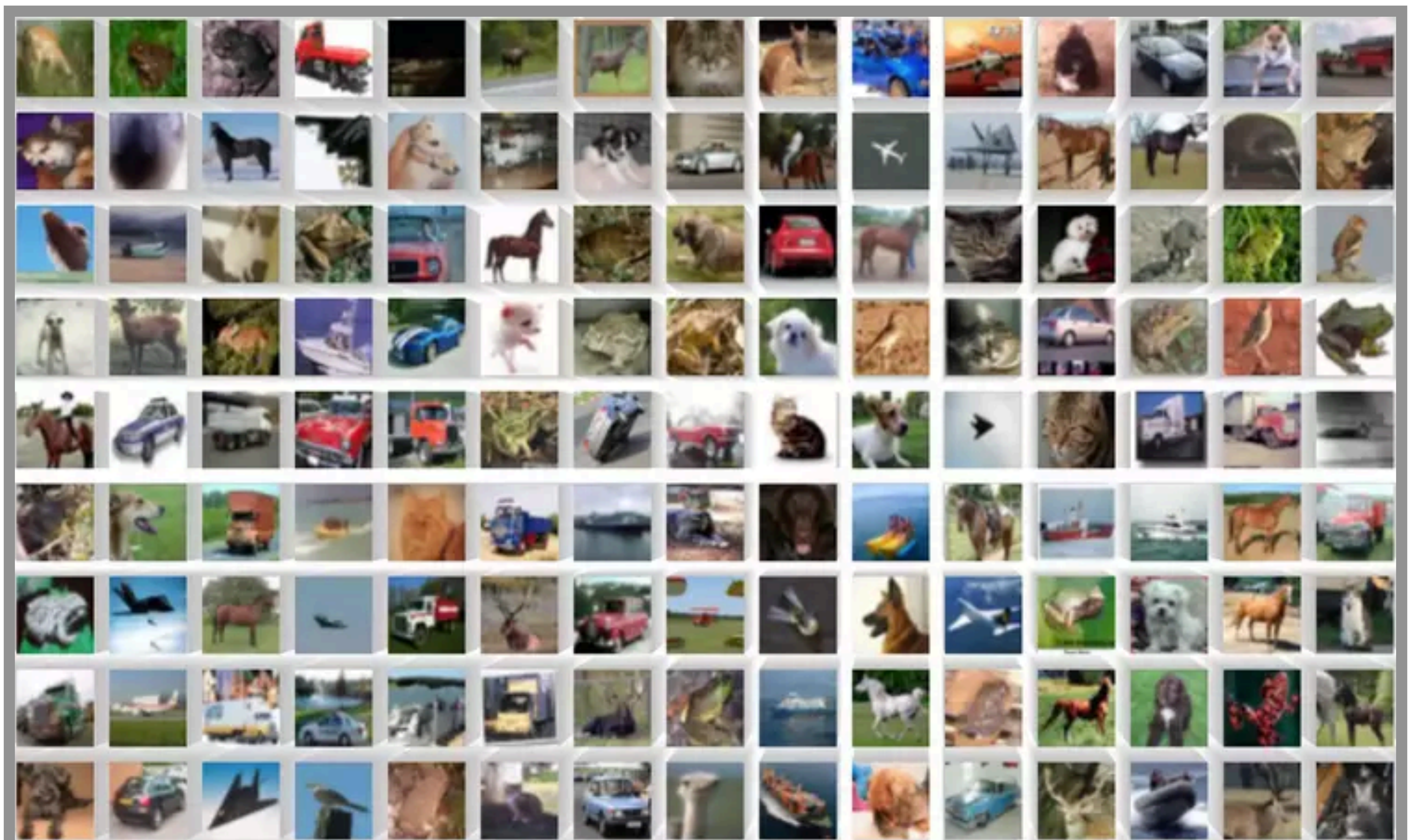
hate speech detection
softmax
one-vs-all (ours)
Expected accuracy
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.5
1.0
Confidence
ECE=6.96%
ECE=0.77%
Gap
Accuracy



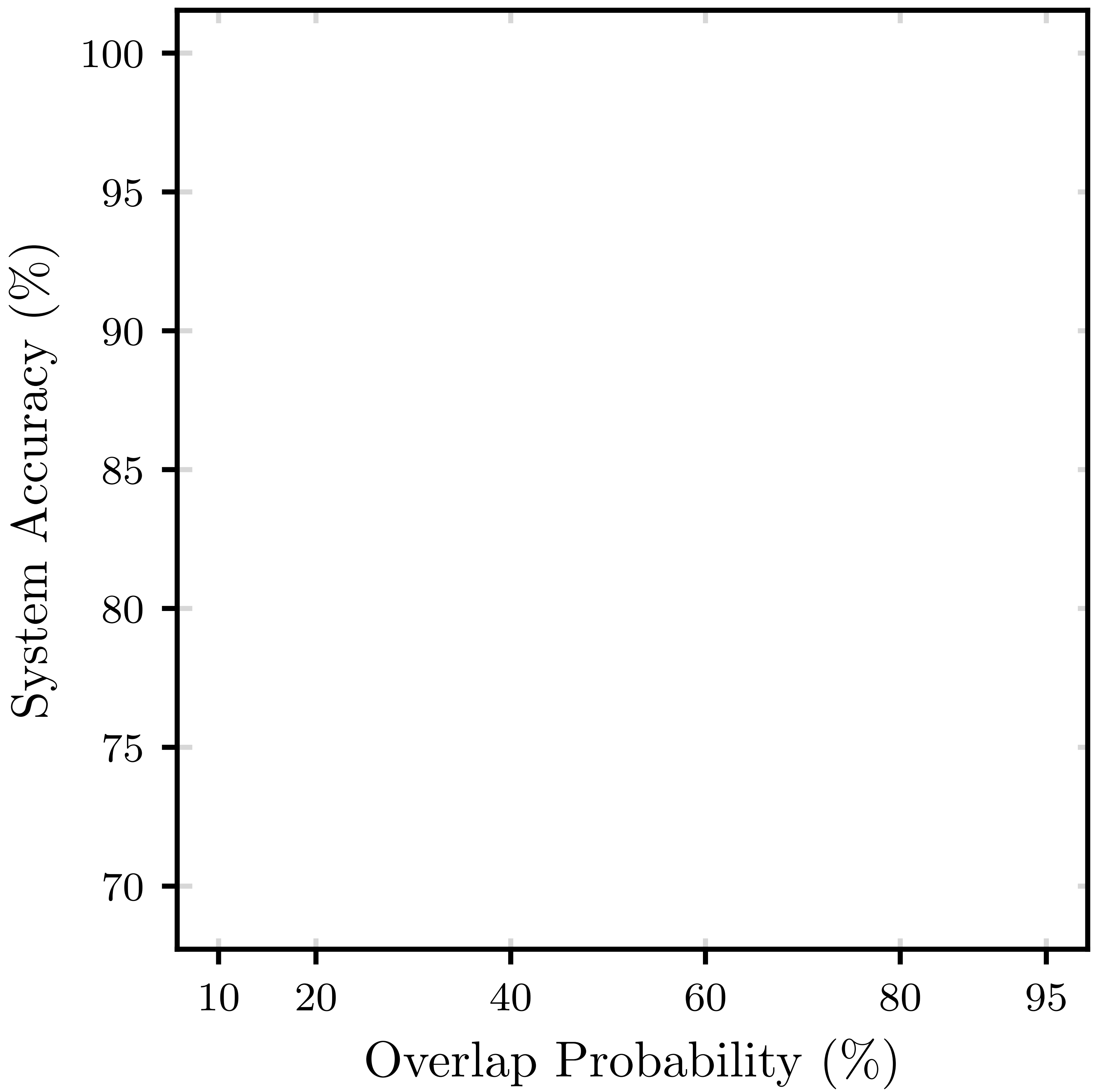
Hate Speech
System Accuracy (%)
Number of Experts
OvA
Softmax
MoE
One Classifier
Best Expert
Hate Speech
ECE (%)
Number of Experts
OvA
Softmax
MoE

conformal: downstream performance

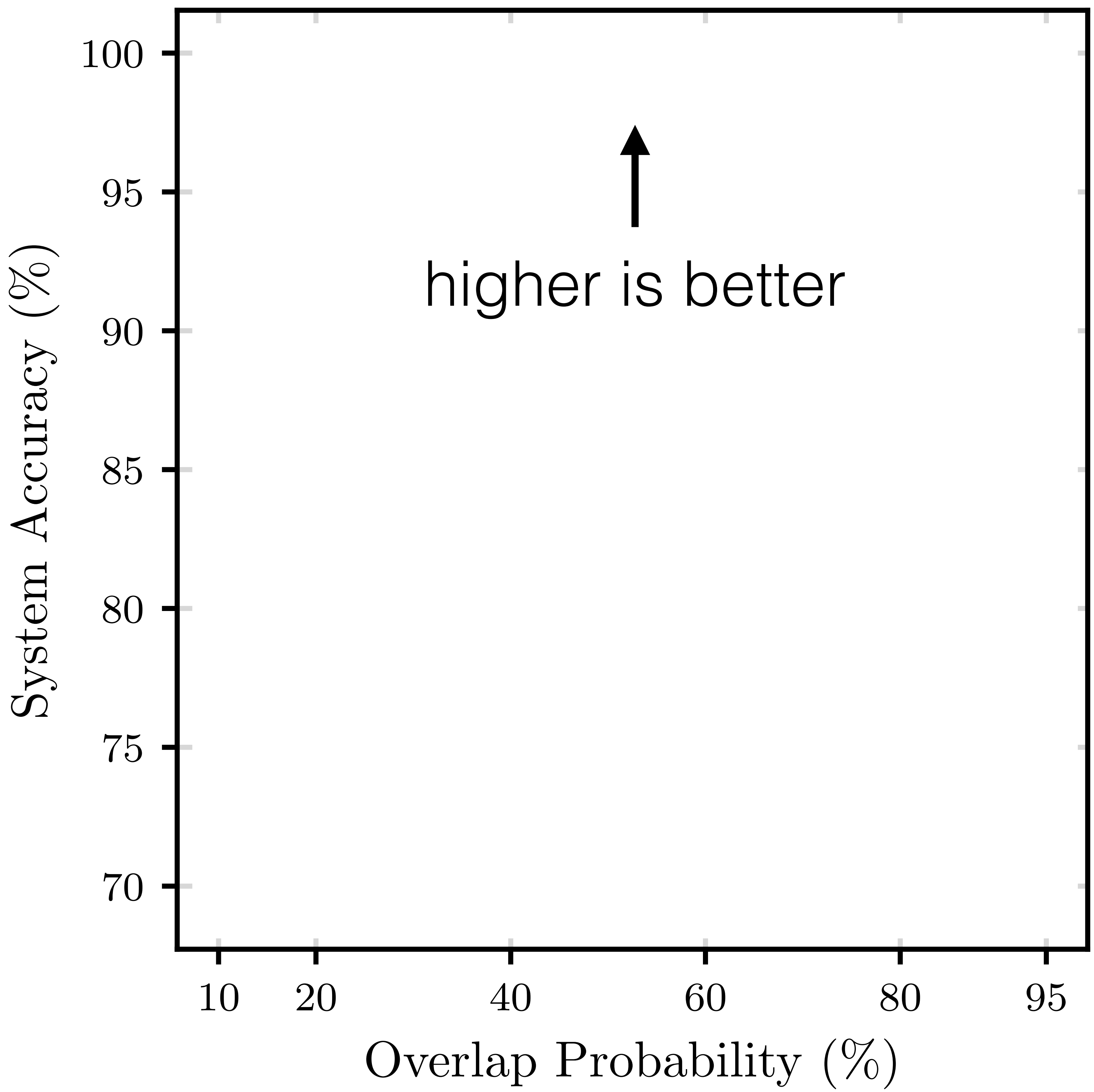
CIFAR-10



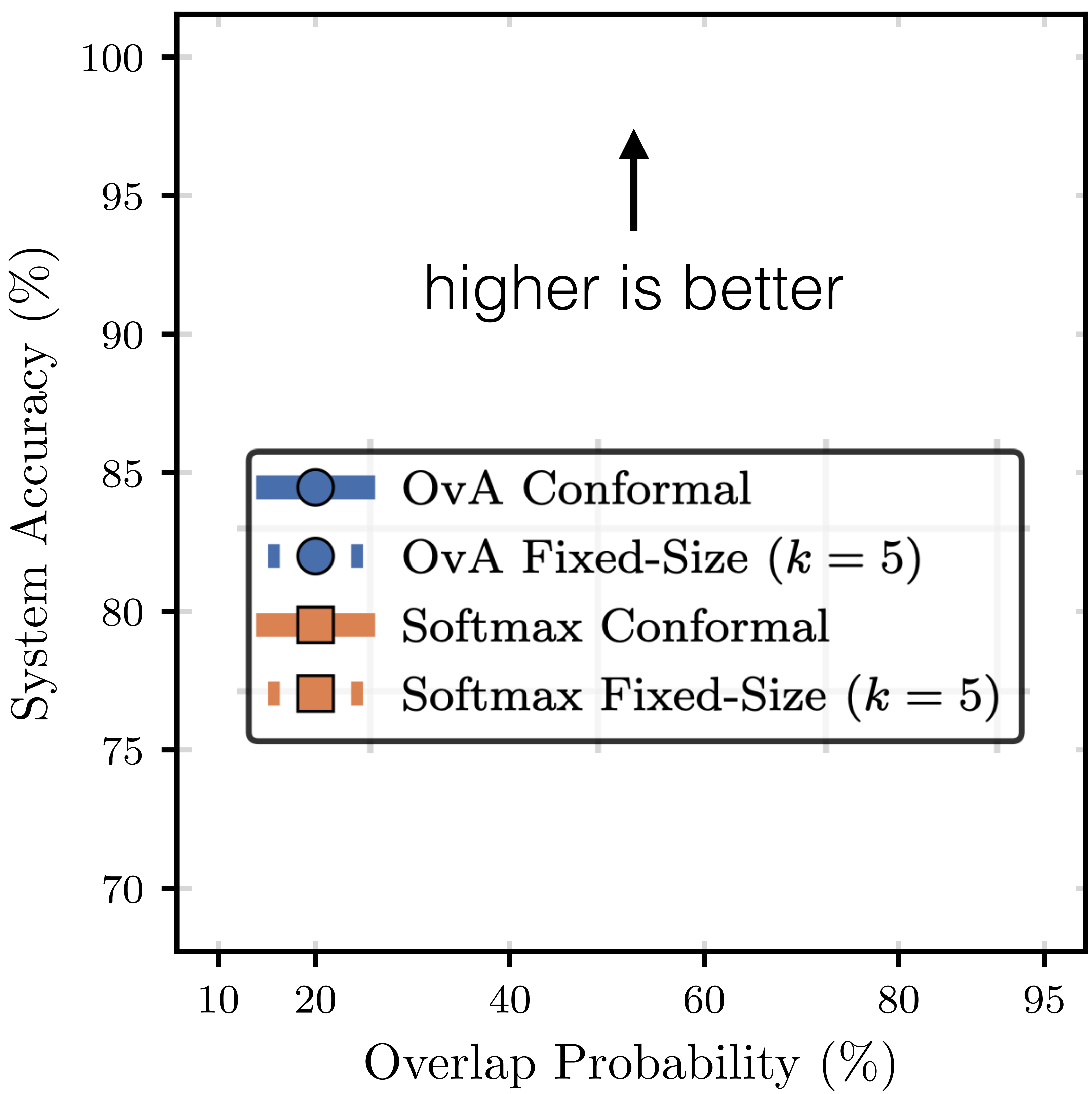
downstream performance



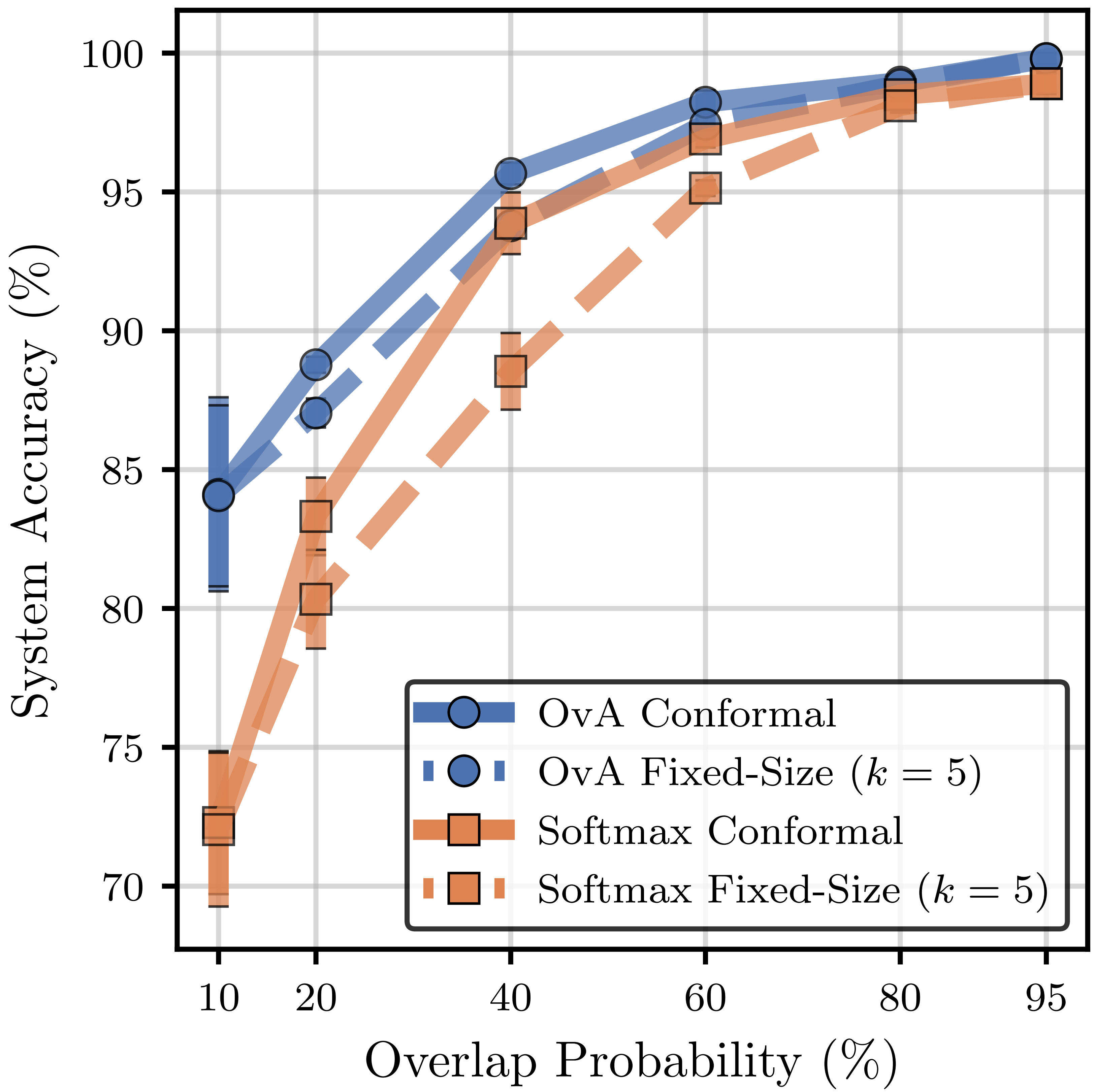
downstream performance



downstream performance



downstream performance



simulated experts:

Table 2: HAM10000 experts configuration.

	Expert configuration	p_{in} [%]	p_{out} [%]	Diagnostic Category [in]
1	Random Expert	-	-	[nv, bkl, df, vasc, mel, bcc, akiec]
2	Dermatologist for malign	25	15	[mel, bcc, akiec]
3	Dermatologist for benign	25	15	[nv, bkl, df, vasc]
4	Specialized dermatologist in nv	50	15	[nv]
5	Specialized dermatologist in vasc	70	15	[vasc]
6	Specialized dermatologist in mel	75	15	[mel]
7	Dermatologist for benign	75	25	[nv, bkl, df, vasc]
8	MLP Mixer	-	-	[nv, bkl, df, vasc, mel, bcc, akiec]
9	Experienced dermatologist	80	50	[nv, bkl, df, vasc, mel, bcc, akiec]
10	Experienced dermatologist	80	60	[nv, bkl, df, vasc, mel, bcc, akiec]

simulated experts:

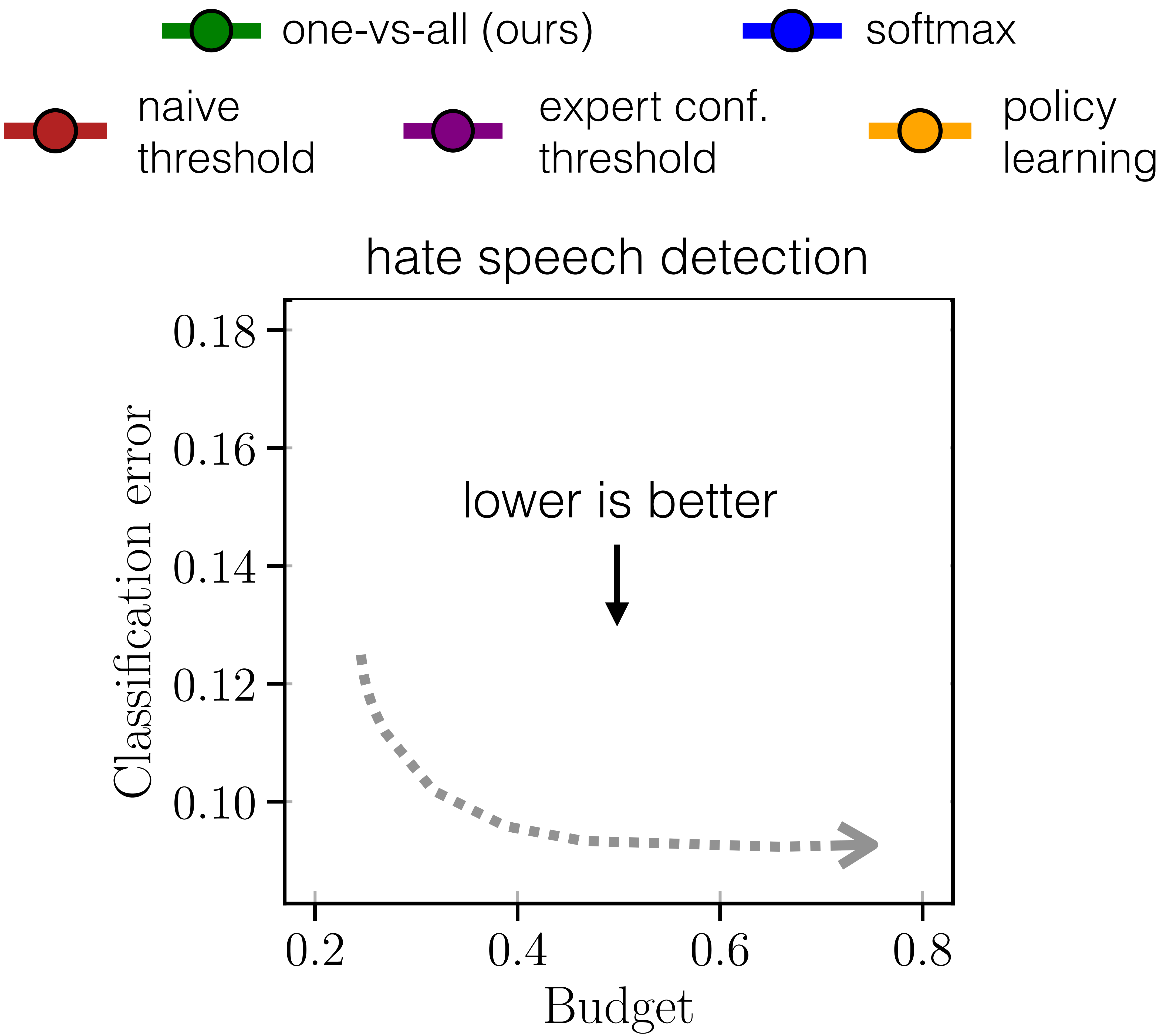
Table 1: Hate Speech and Galaxy-Zoo experts configuration.

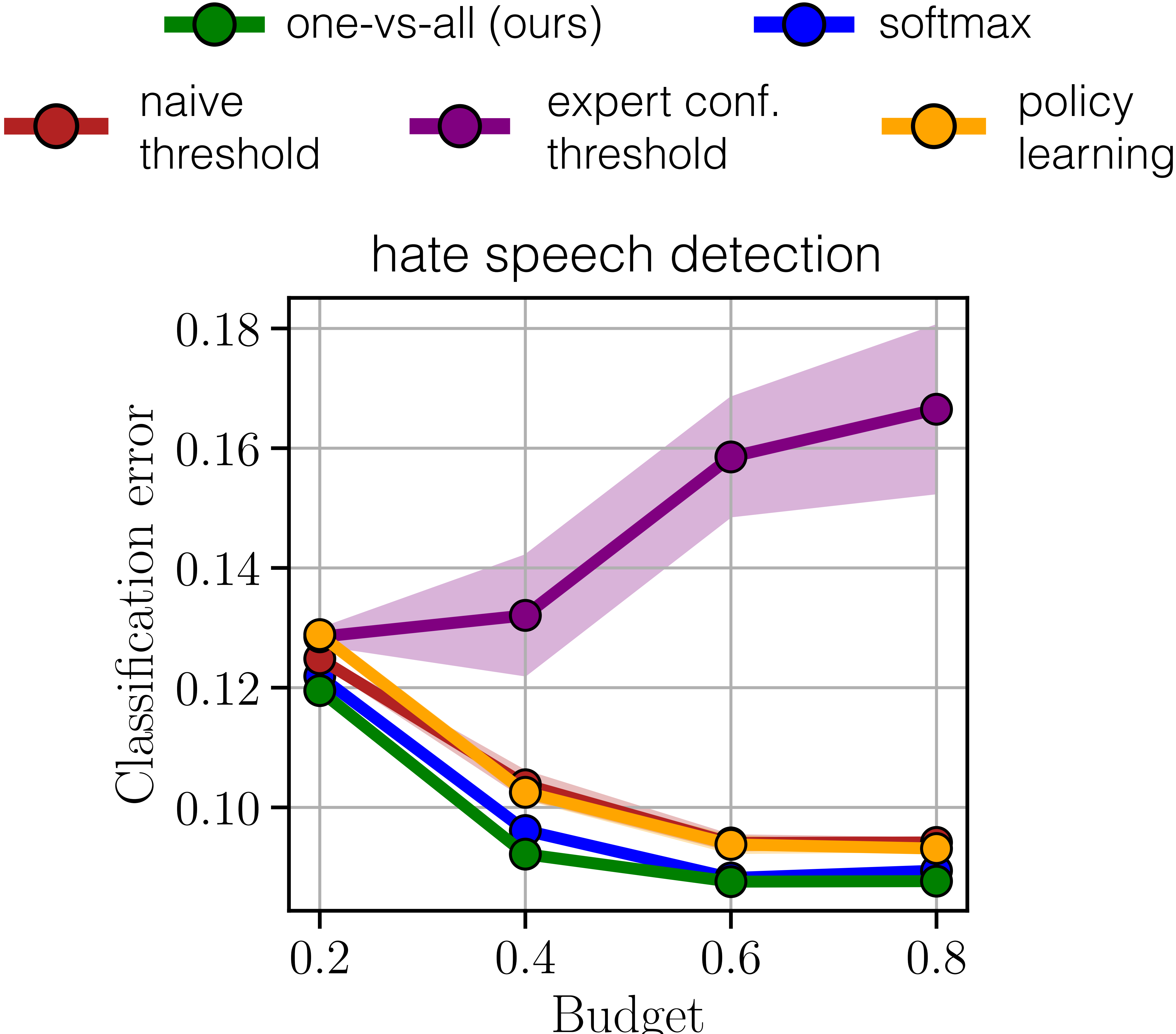
	Expert configuration	$\texttt{p}_{\texttt{flip}}$[%]	$\texttt{p}_{\texttt{annotator}}$[%]
1	Random Expert	-	-
2	Probabilistic Expert	-	10
3	Flipping Human Expert	50	-
4	Probabilistic Expert	-	75
5	Flipping Human Expert	30	-
6	Flipping Human Expert	20	-
7	Probabilistic Expert	-	85
8	Human Expert	-	-
9	Probabilistic Expert	-	50
10	Human Expert	-	-

one-vs-all (ours)
softmax
naive threshold
expert conf. threshold
policy learning

hate speech detection

[Davidson et al., ICWSM 2017]


one-vs-all (ours)
softmax
naive threshold
expert conf. threshold
policy learning
hate speech detection
lower is better
0.18
0.16
0.14
0.12
0.10
Classification error
0.2
0.4
0.6
0.8
Budget


one-vs-all (ours)
softmax
naive threshold
expert conf. threshold
policy learning
hate speech detection
Classification error
0.18
0.16
0.14
0.12
0.10
0.2
0.4
0.6
0.8
Budget

conformal inference: train-time

expert #1

expert #2

expert #3

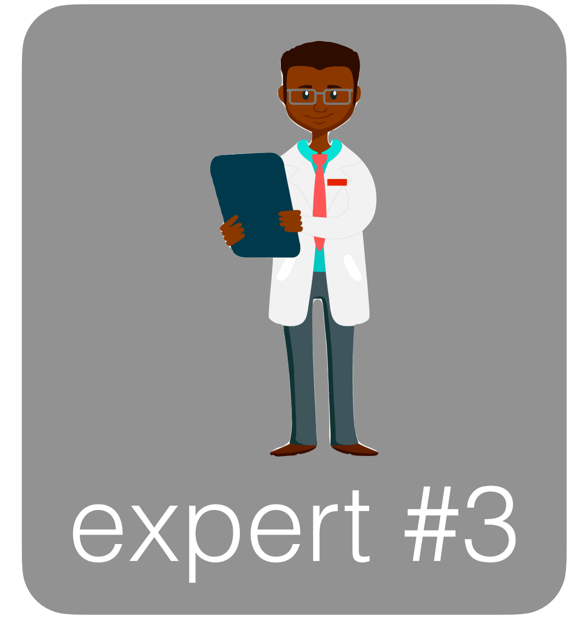
conformal inference: train-time

$h_{\perp,1}(x)$ $h_{\perp,2}(x)$ $h_{\perp,3}(x)$

conformal inference: train-time

$\mathbf{h}_{\perp,1}(\mathbf{x})$

$\mathbf{h}_{\perp,2}(\mathbf{x})$



$\mathbf{h}_{\perp,3}(\mathbf{x})$

using validation data, compute the (1-α)-quantile of a conformity statistic:

$$\hat{q}_{1-\alpha}$$

conformal inference: test-time

$h_{\perp,1}(x)$

$h_{\perp,2}(x)$

$h_{\perp,3}(x)$

conformal inference: test-time

$$h_{\perp,3}(x) > h_{\perp,1}(x) > h_{\perp,2}(x)$$

conformal inference: test-time

$$\mathsf{h}_{\perp,3}(\mathsf{x}) \quad > \quad \mathsf{h}_{\perp,1}(\mathsf{x}) \quad > \quad \mathsf{h}_{\perp,2}(\mathsf{x})$$

$$\mathsf{C}(\mathsf{x}) = \left\{ \qquad\qquad \right\}$$

check if:

$$\sum_{e \in \mathsf{C}(\mathsf{x})} \mathsf{h}_{\perp,e}(\mathsf{x}) \overset{?}{\geq} \hat{q}_{1-\alpha}$$

conformal inference: test-time

expert #1

expert #2

$$\mathsf{h}_{\perp,3}(\mathsf{x}) \quad > \quad \mathsf{h}_{\perp,1}(\mathsf{x}) \quad > \quad \mathsf{h}_{\perp,2}(\mathsf{x})$$

$\mathsf{C}(\mathsf{x}) = \Big\{$  $\Big\}$

check if:

$$\mathsf{h}_{\perp,3} \overset{?}{\geq} \hat{q}_{1-\alpha}$$

conformal inference: test-time

$$h_{\perp,3}(x) > h_{\perp,1}(x) > h_{\perp,2}(x)$$

$$C(x) = \{ \quad \}$$

check if:

$$h_{\perp,3} \geq \hat{q}_{1-\alpha}$$

conformal inference: test-time

$$h_{\perp,3}(x) > h_{\perp,1}(x) > h_{\perp,2}(x)$$

$C(x) = \{$  ,  $\}$

check if:

$$h_{\perp,3} + h_{\perp,1} \overset{?}{\geq} \hat{q}_{1-\alpha}$$

conformal inference: test-time

$$h_{\perp,3}(x) > h_{\perp,1}(x) > h_{\perp,2}(x)$$

$C(x) = \{$ ,  $\}$

check if:

$$h_{\perp,3} + h_{\perp,1} \geq \hat{q}_{1-\alpha}$$

Estimating $\mathbb{P}(\mathbf{m} = \mathbf{y} \mid \mathbf{x})$