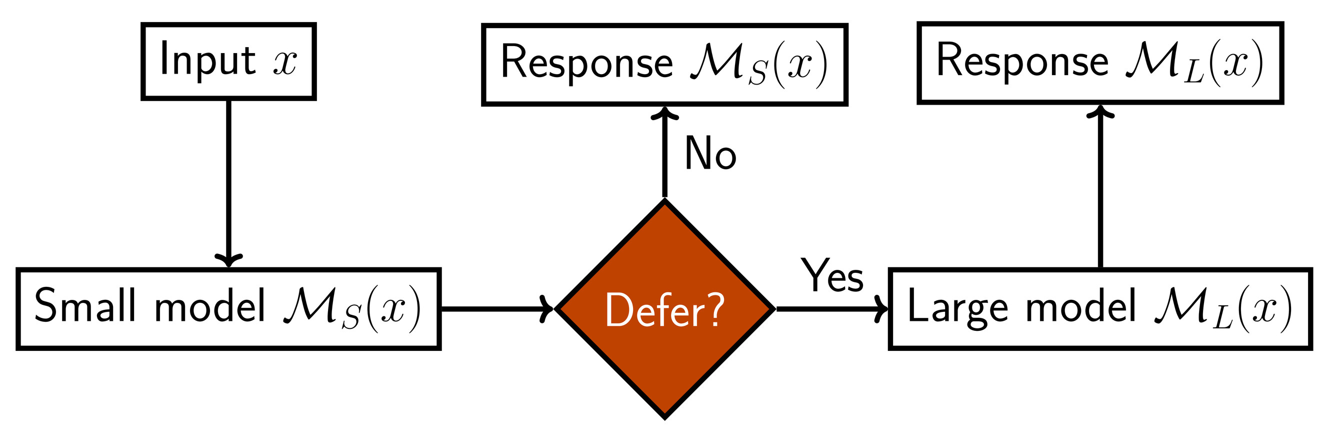
Gatekeeper: Improving Model Cascades Through Confidence Tuning

Stephan Rabanser, *Princeton University* **Nathalie Rauschmayr**, **Achin Kulshrestha**, **Petra Poklukar**, **Wittawat Jitkrittum**, **Sean Augenstein**, **Congchao Wang**, **Federico Tombari**, *Google Research*

Main Contribution

We introduce a new loss function that calibrates smaller models in cascade setups to confidently handle easy examples while at the same time deferring more complex queries.

Cascading Overview



Assumption

We assume that $\mathcal{M}_L$ **dominates** $\mathcal{M}_S$ with high probability:

$$\Pr_{(x,y)\sim\mathcal{D}}\left[\mathcal{M}_L(x) \neq y \ \wedge\ \mathcal{M}_S(x) = y\right] \leq \delta, \qquad (1)$$

with $\delta \ll 1$. This "almost-always" dominance, supported by scaling-law trends, implies that deferring from $\mathcal{M}_S$ to $\mathcal{M}_L$ cannot hurt accuracy in expectation, while still allowing rare counter-examples where the small model outperforms the large model.

Question

Can we optimize the small model to separate correct from incorrect predictions?

A Workflow for Better Cascading

1. **Standard training**: We begin with an $\mathcal{M}_S$ that has already been trained on the tasks it is intended to perform upon deployment.

2. **Finetuning with Gatekeeper**: We introduce a correctness-aware loss to fine-tune $\mathcal{M}_S$ for improved confidence calibration.

3. **Deferral via uncertainty thresholding**: Given a deferral function $g : \mathbb{R}^D \to \mathbb{R}$ and a targeted acceptance threshold $\tau \in \mathbb{R}$:

$$(\mathcal{M}_S, \mathcal{M}_L, g)(x) = \begin{cases} \mathcal{M}_S(x) & g(x) \geq \tau \\ \mathcal{M}_L(x) & \text{otherwise.} \end{cases}$$

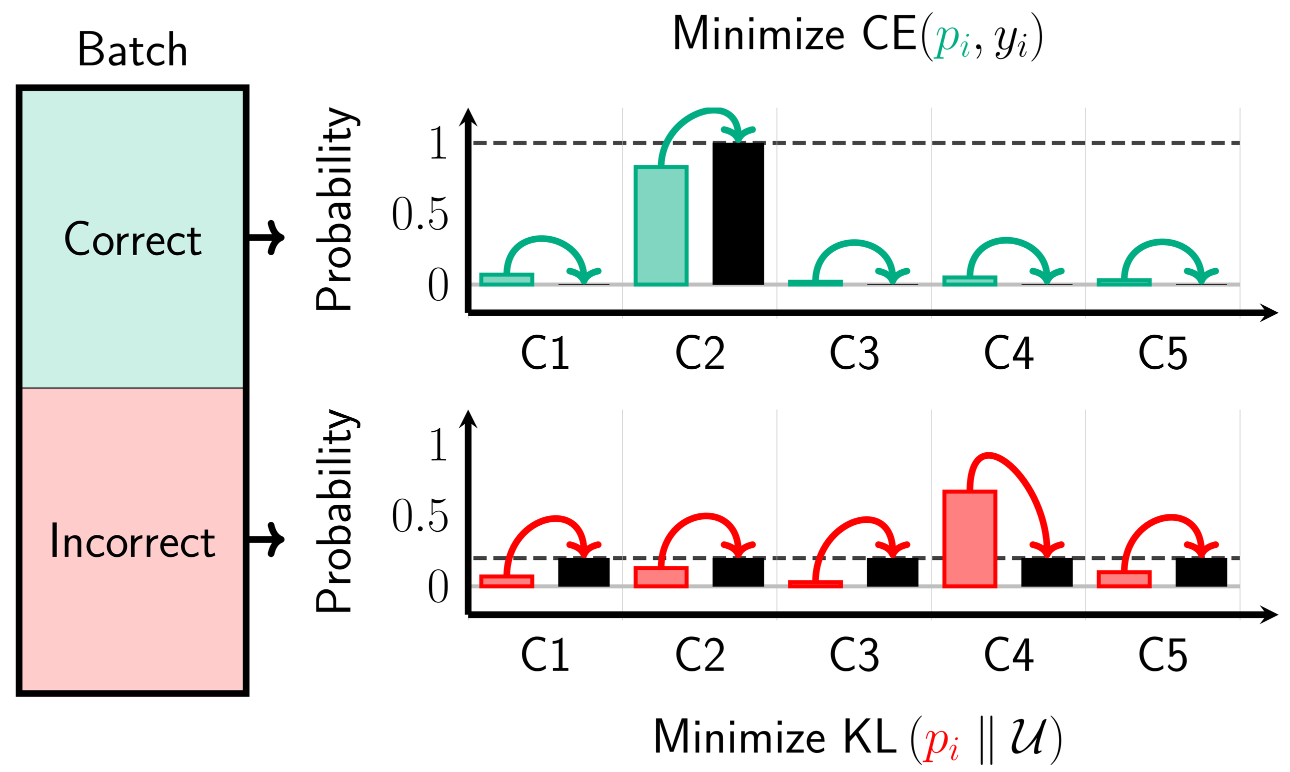
Classification: $g_{\text{CL}}(x) = \max_{1\leq c \leq C} p(y = c \mid x)$.
Sequence: $g_{\text{NENT}}(x) = \frac{1}{T}\sum_{t=1}^{T}\sum_{c=1}^{C} p(y_t = c \mid x) \log p(y_t = c \mid x)$

Higher values of g_{CL} or g_{NENT} indicate lower predictive uncertainty.

Confidence Calibration With Gatekeeper

Idea

Fine-tune the small model $\mathcal{M}_S$ by regularizing wrong predictions to a uniform distribution $\mathcal{U}$.



$$\mathcal{L} = \alpha\mathcal{L}_{\text{corr}} + (1-\alpha)\mathcal{L}_{\text{incorr}}$$

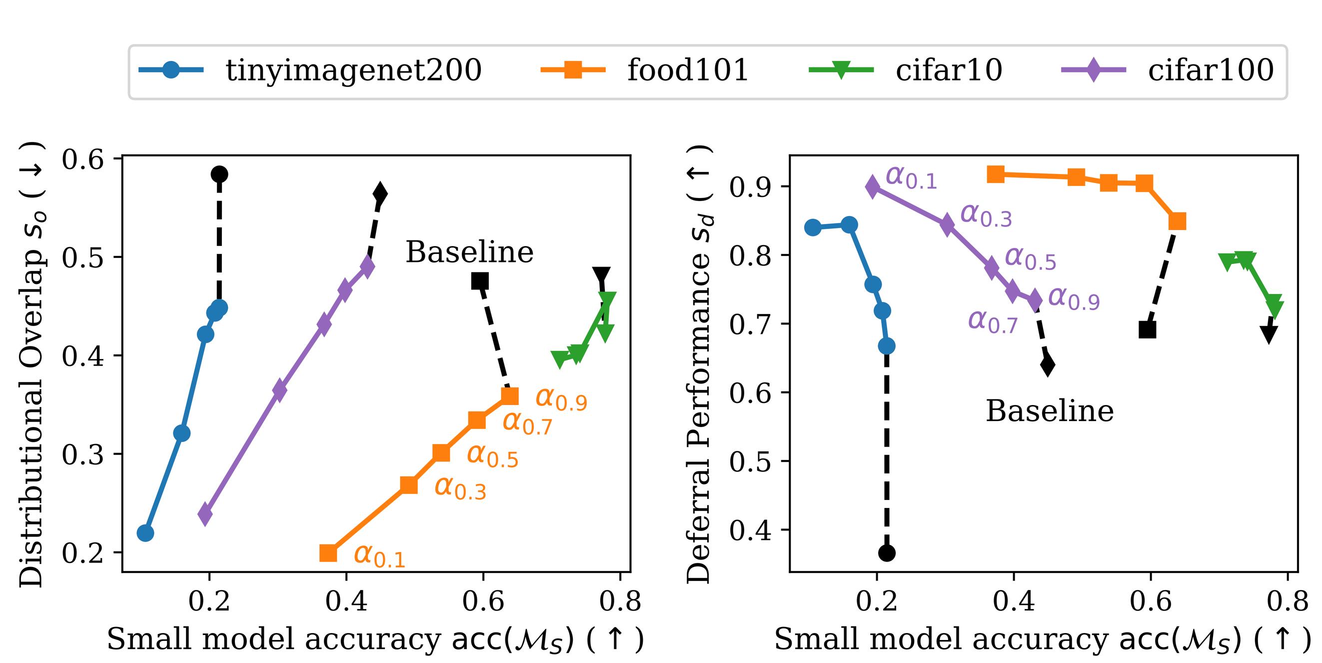
↓ Low α emphasizes confidence calibration of incorrect data points.
↑ High α emphasizes maintaining high utility over full distribution.

$$\mathcal{L}_{\text{corr}} = \frac{1}{N}\sum_{\substack{i=1\\t=1}}^{N,T} \mathbb{1}\{y_{i,t} = \hat{y}_{i,t}\}\,\text{CE}(p_{i,t}(x_i), y_{i,t})$$

$$\mathcal{L}_{\text{incorr}} = \frac{1}{N}\sum_{\substack{i=1\\t=1}}^{N,T} \mathbb{1}\{y_{i,t} \neq \hat{y}_{i,t}\}\,\text{KL}\left(p_{i,t}(x_i) \parallel \mathcal{U}\right)$$

Observation

There is a tradeoff between deferral performance and overall accuracy. $\mathcal{M}_S$ effectively unlearns handling hard data points.



Empirical Results Across Classification, Language, and Multi-Modal Models