Data Shunt: Collaboration of Small and Large Models for Lower Costs and Better Performance

Dong Chen[1], Yueting Zhuang[1,*], Shuo Zhang[1], Jinfeng Liu[2], Su Dong[2], Siliang Tang[1]
[1]Zhejiang University, [2]Ant Group

The Thirty-Eighth AAAI Conference on Artificial Intelligence (**AAAI-24**)

Xincao Xu

Shenzhen Institute for Advanced Study, UESTC

September 5, 2024

Research Team

Yueting Zhuang
Professor | Doctoral supervisor
Former Dean of College of CS
Zhejiang University

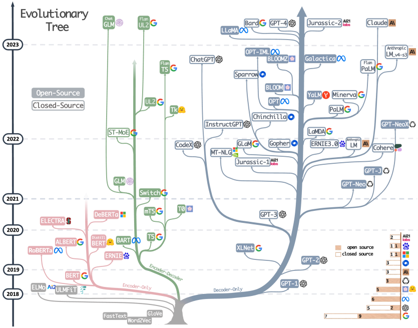
Research Interests

- Artificial Intelligence
- Cross-media Computing
- Multimedia Retrieval

Awards and Honors

- Distinguished Young Scholars
- "Chang Jiang Scholars Program" Professor
- Fellow of Chinese Association for Artificial Intelligence (CAAI)
- Fellow of China Society of Image and Graphics

Advantages of Pretrained Large Models (PLMs)



Growing attention on PLMs due to their versatility across tasks

- *Outstanding Performance*: LLMs like GPT show exceptional ability in text-related tasks
- *Multimodal Capabilities*: Models like Flamingo and BLIP-2 expand LLMs to handle vision tasks
- *Widespread Applications*: Models like ChatGPT are transforming fields like coding, education, and healthcare

Picture Source: Yang, Jingfeng, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Shaochen Zhong, Bing Yin, and Xia Hu. "Harnessing the power of llms in practice: A survey on chatgpt and beyond." ACM Transactions on Knowledge Discovery from Data 18, no. 6 (2024): 1-32.

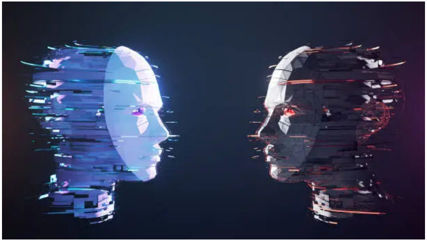
Large Model vs. Small Model

Large Model

- High computational demand
- Impractical for deployment on many devices
- Costly interface access
- Higher accuracy on general tasks
- Greater overhead for deployment and switching

Small Model

- Lower computational demand
- Easily deployable on resource-constrained devices
- More affordable access
- May outperform large models on *specific data distributions*
- Less overhead, faster switching



Sample Classification

Easy Samples

- Small models fit well, representing the majority of training data
- Small models are efficient for these samples
- Less computational cost with small models
- Risk of overfitting on limited datasets

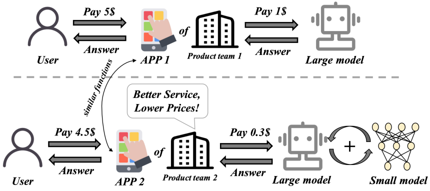
Hard Samples

- Large models handle challenges, e.g., long-tail distributions, boundary cases
- Large models offer higher accuracy on difficult data
- Higher computational demand, but necessary for complex cases
- Better generalization to out-of-distribution and challenging inputs

Question

How can a collaborative paradigm be introduced to reduce large model calls and enhance performance?

Innovative Methods: Data Shunt Collaborative Paradigm



- Upper: Only use large models to support their applications
- Lower: Decrease costs with collaboration of large and small models

Data Routing Based on Confidence

- Small models determine if data should be processed by large models or handled independently

Prompt Pruning

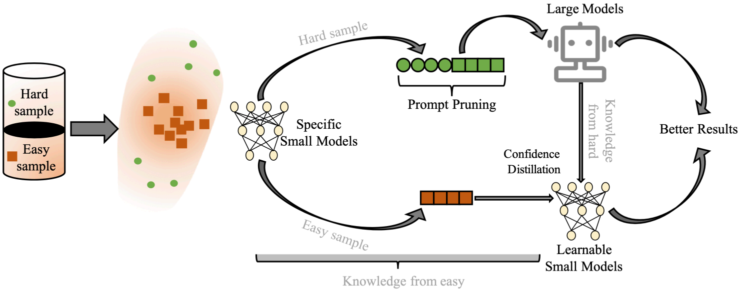
- Utilizes small models to refine prompts for large models, improving prediction accuracy

2-Stage Confidence Distillation

- Enables small models to learn iteratively from large models, mitigating catastrophic forgetting

1. Introduction
2. **Methodology**
3. Experiments
4. Conclusion

Overview



Determine the shunt threshold by evaluating small models confidence with training set

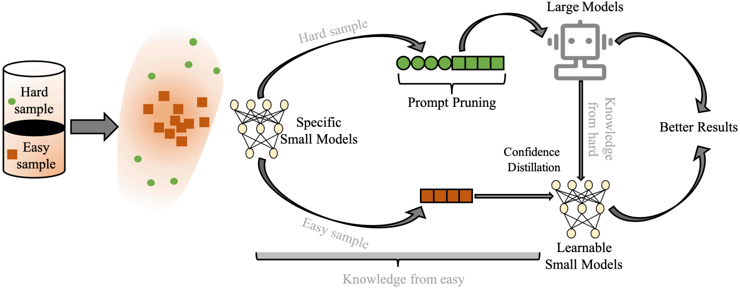
Hard Examples

- Challenging for small models
- Deviating from main data distributions
- Lying at category boundaries

Easy Examples

- Majority of training data
- Fitting well with small models
- Being easier to learn and predict

Workflow



Small Models for Large Models, Prompt Pruning (PP)

- Identify small model strengths
- Introduce prompt pruning
- Craft prompts using small models
- Refine large model predictions

Large Models for Small Models, Confidence Distillation

- Establish two small model versions
- Specific and learnable small models
- Confidence-based distillation
- Balance knowledge acquisition

Small Models for Large Models, Prompt Pruning (PP)

Prediction Confidence Computation

- Subjecting the output of a trained small model F_{small} to a softmax operation for a given input x

$$C_s = \frac{e^{z_i}}{\sum e^{z_d}}, \quad z_i \in F_s(x) \tag{1}$$

- where $F_s(x)$ represents the output (logits) of the small model F_{small} for input x
- z_i is one of the logits, corresponding to a possible class (e.g., "cat", "dog", etc.)

Enhancing Large Model Predictions with Small Model Confidence

- Small model excels at distinguishing specific classes (e.g., cats).
- Unable to recognize other animals (e.g., dogs, tigers), but confidently identifies them as not cats (low confidence).
- Use small model predictions to guide large models by refining their prediction space.
- Improves large model performance through enhanced focus on relevant categories.

Small Models for Large Models, Prompt Pruning (PP)

Incorporating Predictions into the Prompts

- To refine the prediction space and enhance the performance

Example

A prompt of PP for image classification task: *"This is a photo of a label with probability C_s"*

PP uses small model confidence as prior knowledge in prompts.

Soft Prompt

- Adds probability to classes small models excel in.
- Large models ignore irrelevant classes, improving accuracy.

Hard Prompt

- Directly removes classes small models excel in.
- Reduces prediction space, increasing accuracy for large models.

Small Models for Large Models, Prompt Pruning (PP)

Theoretical Analysis of Soft Prompts Enhancing Large Model Performance

- X: Variable of input data
- Y: Variable of small model prediction
- Entropy: To quantify the lower bound of model capability, the lower is better
- $H(X)$: Entropy of the input data
- $H(Y)$: Entropy of the prediction
- $H(X|Y) = H(\hat{X})$: Entropy of the input data with soft prompt

$$\begin{aligned} H(X) - H(\hat{X}) &= \sum_{x \in X} \sum_{y \in Y} p(x,y) \log_2 \frac{p(x,y)}{p(x)p(y)} \\ &\geq \left[\sum_{x \in X} \sum_{y \in Y} p(x,y) \right] \log_2 \frac{\sum_{x \in X} \sum_{y \in Y} p(x,y)}{\sum_{x \in X} \sum_{y \in Y} p(x)p(y)} = 0 \end{aligned} \tag{2}$$

Small Models for Large Models, Prompt Pruning (PP)

Definition of Entropy

- For a random variable X, its entropy $H(X)$ is a measure of uncertainty

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x) \tag{3}$$

- For another random variable Y, the entropy $H(Y)$ is

$$H(Y) = -\sum_{y \in Y} p(y) \log_2 p(y) \tag{4}$$

- The joint entropy $H(X, Y)$, which quantifies the uncertainty of both X and Y together

$$H(X, Y) = -\sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 p(x, y) \tag{5}$$

Small Models for Large Models, Prompt Pruning (PP)

Definition of Conditional Entropy

- Conditional entropy $H(X|Y)$ represents the uncertainty of X given Y

$$H(X|Y) = -\sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 p(x|y) \tag{6}$$

- Using the relationship between joint and conditional probabilities, $p(x, y) = p(y)p(x|y)$

$$\begin{aligned} H(X|Y) &= -\sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 \frac{p(x, y)}{p(y)} \\ &= -\sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 p(x, y) + \sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 p(y) \\ &= -\sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 p(x, y) + \sum_{y \in Y} p(y) \log_2 p(y) \\ &= H(X, Y) - H(Y) \end{aligned} \tag{7}$$

Small Models for Large Models, Prompt Pruning (PP)

Deriving the Entropy Difference

$$
\begin{aligned}
H(X) - H(\hat{X}) &= H(X) - H(X|Y) \\
&= H(X) - (H(X,Y) - H(Y)) \\
&= H(X) + H(Y) - H(X,Y) \\
&= -\sum_{x\in X} p(x)\log_2 p(x) - \sum_{y\in Y} p(y)\log_2 p(y) + \sum_{x\in X}\sum_{y\in Y} p(x,y)\log_2 p(x,y) \\
&= -\sum_{x\in X}\sum_{y\in Y} p(x,y)\log_2 p(x) - \sum_{y\in Y}\sum_{x\in X} p(x,y)\log_2 p(y) \\
&+ \sum_{x\in X}\sum_{y\in Y} p(x,y)\log_2 p(x,y) \\
&= \sum_{x\in X}\sum_{y\in Y} p(x,y)\log_2 \frac{p(x,y)}{p(x)p(y)}
\end{aligned} \tag{8}
$$

Small Models for Large Models, Prompt Pruning (PP)

Definition of Kullback-Leibler (KL) Divergence

- KL divergence $KL(P, Q)$ is a measure of the "distance" between two probability distributions P and Q

$$KL(P, Q) = \sum_{x} p(x) \log_2 \frac{p(x)}{q(x)} \tag{9}$$

- KL divergence between the joint distribution $p(x, y)$ and the product of the marginal distributions $p(x)p(y)$

$$KL(p(x, y), p(x)p(y)) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 \frac{p(x, y)}{p(x)p(y)} \tag{10}$$

- KL divergence is always non-negative based on Jensen's Inequality and equals zero only when $p(x, y) = p(x)p(y)$, which means X and Y are independent

$$H(X) - H(\hat{X}) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 \frac{p(x, y)}{p(x)p(y)} \geq 0 \tag{11}$$

Small Models for Large Models, Prompt Pruning (PP)

Theoretical Analysis of Hard Prompts Enhancing Large Model Performance

- Entropy of the prediction for large model

$$H(C_l) = -\sum_{i=1}^{N} c_i \log c_i \tag{12}$$

- where N is the number of possible candidates (e.g., classes, categories).
- c_i represents the probability of the i-th candidate in C_l
- C_l denotes the set of candidate classes for large model predictions
- The sum of all probabilities must equal 1

$$\sum_{i=1}^{N} c_i - 1 = 0 \tag{13}$$

- A classic setup for using the **Lagrange multiplier method** to find the maximum entropy

Small Models for Large Models, Prompt Pruning (PP)

Setting Up the Lagrange Multiplier

- Lagrange function G is formulated to include both the entropy function and the constraint

$$G(c_1, c_2, \ldots, c_N, \lambda) = -\sum_{i=1}^{N} c_i \log c_i + \lambda \left(\sum_{i=1}^{N} c_i - 1 \right) \tag{14}$$

- where λ is the Lagrange multiplier
- The first term represents the entropy to be maximized.
- The second term enforces the constraint $\sum_{i=1}^{N} c_i = 1$.

Partial Differentiation

- Differentiation with respect to c_i

$$\frac{\partial G}{\partial c_i} = -\log c_i - 1 + \lambda \tag{15}$$

- Differentiation with respect to λ

$$\frac{\partial G}{\partial \lambda} = \sum_{i=1}^{N} c_i - 1 \tag{16}$$

Small Models for Large Models, Prompt Pruning (PP)

Solving for c_i

- By setting $\frac{\partial G}{\partial c_i} = 0$ and $\frac{\partial G}{\partial \lambda} = 0$

$$\begin{cases} \frac{\partial G}{\partial c_i} = -\log c_i - 1 + \lambda = 0 \\ \frac{\partial G}{\partial \lambda} = \sum_{i=1}^{N} c_i - 1 = 0 \end{cases} \tag{17}$$

- Since c_i is the same for all i, we can express the probabilities $c_1 = c_2 = \cdots = c_N$

$$\begin{cases} c_i = e^{\lambda - 1} \\ c_i = \frac{1}{N} \end{cases} \rightarrow c_i = \frac{1}{N} \tag{18}$$

Maximum Entropy for Large Models

- Substituting $c_i = \frac{1}{N}$ into the entropy formula
- Maximum entropy occurs when all N candidates are equally probable

$$H(C_I) = -\sum_{i=1}^{N} \frac{1}{N} \log \frac{1}{N} = \log N \tag{19}$$

Small Models for Large Models, Prompt Pruning (PP)

Performing Hard Prompt with Fewer Candidates

- When applying a hard prompt, the number of candidates reduces to $M < N$

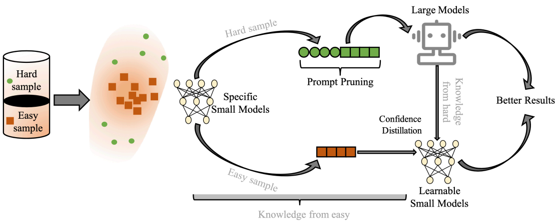
$$H(\hat{C}_l) = \log M \tag{20}$$

- Since $M < N$

$$\log M < \log N \rightarrow H(\hat{C}_l) < H(C_l) \tag{21}$$

- When the number of candidates is reduced due to the hard prompt, the new entropy $H(\hat{C}_l)$ is lower
- The model has less uncertainty in its predictions
- The lower bound on the large models performance increases

Large Models for Small Models, 2-Stage Confidence Distillation (2CD)



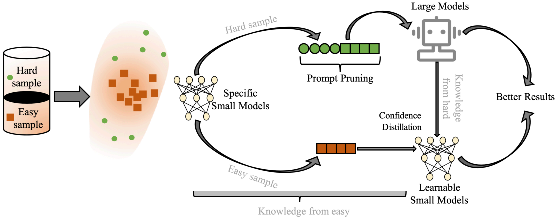
Key Idea

- Large models help small models by distilling knowledge that small models lack
- Expanding small model knowledge reduces the need to invoke large models

Issues

- Small models may forget original distributions after distillation
- Large models can degrade small models performance if incorrect knowledge is distilled

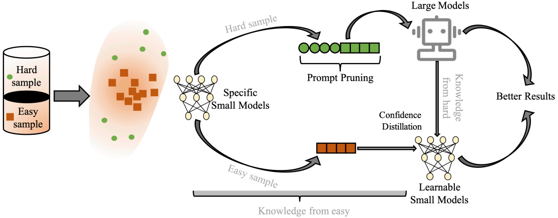
Large Models for Small Models, Confidence Distillation



Solution: 2-Stage Confidence Distillation (2CD)

- Maintain original small models (specific small models) without large model influence
- Create duplicated small models (learnable small models) to receive knowledge
- Learnable small models learn from both large and specific small models
- High confidence in large models = Learnable small models acquire knowledge
- High confidence in specific small models = Learnable small models prioritize them to avoid incorrect knowledge

Large Models for Small Models, Confidence Distillation



Compute Small Model Output

- Given input data x, the confidence C_{s1} is computed from the specific small model
- When it is lower than shunt threshold $C_{s1} < \delta$

Compute Large Model Prediction

- Prediction confidence C_l from the large model $F_l(x)$

$$C_l = \frac{e^{z_i}}{\sum e^{z_d}}, \quad z_i \in F_l(x) \tag{22}$$

- where z_i is the output of the large model for the input x

Large Models for Small Models, Confidence Distillation

Stage 1 Knowledge Distillation

- When the prediction confidence C_l is greater than the threshold δ
- Loss function L_{ls} is defined using the KL divergence

$$L_{ls} = KL(F_{s2}(x), C_l) \tag{23}$$

Stage 2 Knowledge Distillation

- To mitigate the impact of distorted knowledge from the large model
- Select samples where $C_{s1} > \delta$ to perform knowledge distillation

$$L_{s1s2} = KL(F_{s2}(x), C_{s1}) \tag{24}$$

Iterative Process

- Stage 1 and Stage 2 Knowledge Distillations are performed iteratively
- Creating a loop for continuous training and optimization of the small model's performance

Settings

Three Experimental Setups

Modality	Large Model	Small Models	Task	Dataset
Language	ChatGPT	TextCNN, LSTM, Fine-tuned BERT	Sentiment Analysis	Amazon Product Data
Vision	CLIP	ResNet-32	Long-tailed Image Classification	CIFAR-100-LT
Multimodality	BLIP-2 (1.1B)	ResNet-101 (encoder) + LSTM (decoder)	Image Caption Generation	Microsoft COCO

Dataset

- *Amazon Product Data*: 20 categories of product comments with positive or negative sentiment labels
  Dataset split: a) Training set: 2,504,958 samples b) Validation set: 277,508 samples c) Testing set: 309,186 samples
- *CIFAR-100-LT*: 100 categories of color images, with each category comprising 600 images of size 32×32 pixels, totaling 60,000 images
- *Microsoft COCO*: 82,783 images with captions

Data Shunt for Language Modality

Evaluation Metrics: Accuracy and Query (Sample proportion processed by ChatGPT)

Category	Small 1	Large	DS		Category	Small 1	Large	DS	
Games	84.34%	96.22%	96.13%	88.88%	Clothing	85.34%	96.89%	94.28%	84.61%
Kindle	89.05%	95.65%	95.83%	75.88%	Beauty	85.37%	97.20%	94.33%	86.99%
Baby	88.63%	96.41%	95.93%	88.99%	Video	85.37%	92.54%	94.32%	87.28%
Movies	85.37%	93.42%	94.23%	87.68%	Lawn	85.36%	89.36%	94.32%	94.47%
Electronics	85.24%	95.41%	94.67%	88.44%	Home	85.39%	96.28%	94.39%	88.10%
Office	85.23%	95.45%	94.68%	92.12%	Toys	85.41%	96.74%	94.40%	87.80%
CDs	84.68%	95.87%	94.86%	91.99%	Grocery	85.43%	96.73%	94.42%	89.80%
Books	85.26%	93.66%	94.20%	81.88%	Automotive	85.42%	94.69%	94.42%	90.34%
Sports	85.26%	95.06%	94.21%	89.00%	Tools	85.41%	94.49%	94.43%	90.58%
Health	85.24%	95.04%	94.23%	89.49%	Pet Supplies	85.40%	94.03%	94.42%	90.84%
Overall	Small 1: 85.40%, Large: 94.43%, DS: 94.42%				Query	Small 1: 0%, Large: 100%, DS: 84.97%			
Category	Small 2	Large	DS		Category	Small 2	Large	DS	
Games	85.29%	96.22%	96.13%	84.01%	Clothing	86.13%	96.89%	94.31%	74.78%
Kindle	89.74%	95.65%	95.85%	71.73%	Beauty	86.17%	97.20%	94.36%	81.93%
Baby	89.33%	96.41%	95.95%	82.56%	Video	86.18%	92.54%	94.35%	78.15%
Movies	86.27%	93.42%	94.25%	80.82%	Lawn	86.17%	89.36%	94.35%	90.64%
Electronics	86.28%	95.41%	94.69%	83.57%	Home	86.21%	96.28%	94.41%	81.93%
Office	86.26%	95.45%	94.69%	89.14%	Toys	86.22%	96.74%	94.43%	81.05%
CDs	85.70%	95.87%	94.88%	87.78%	Grocery	86.24%	96.73%	94.45%	84.16%
Books	86.05%	93.66%	94.23%	77.59%	Automotive	86.24%	94.69%	94.45%	87.44%
Sports	86.05%	95.06%	94.24%	82.66%	Tools	86.23%	94.49%	94.45%	86.58%
Health	86.03%	95.04%	94.26%	85.12%	Pet Supplies	86.21%	94.03%	94.45%	86.26%
Overall	Smal 2: 86.21%, Large: 94.43%, DS: 94.44%				Query	Small 2: 0%, Large: 100%, DS: 80.00%			
Category	Small 3	Large	DS		Category	Small 3	Large	DS	
Games	90.39%	96.22%	96.15%	36.25%	Clothing	95.63%	96.89%	97.45%	31.10%
Kindle	95.89%	95.65%	97.38%	20.67%	Beauty	92.90%	97.20%	97.24%	30.39%
Baby	92.81%	96.41%	96.26%	32.90%	Video	92.67%	92.54%	96.27%	25.32%
Movies	90.57%	93.42%	94.86%	31.76%	Lawn	84.26%	89.36%	90.63%	48.93%
Electronics	91.76%	95.41%	96.11%	39.56%	Home	93.12%	96.28%	96.73%	33.89%
Office	90.72%	95.45%	95.10%	44.31%	Toys	92.22%	96.74%	96.38%	31.34%
CDs	88.57%	95.87%	95.50%	36.92%	Grocery	92.50%	96.73%	96.66%	32.28%
Books	91.98%	93.66%	95.37%	27.91%	Automotive	91.30%	94.69%	94.69%	33.33%
Sports	93.11%	95.06%	96.02%	34.83%	Tools	91.62%	94.49%	95.38%	37.87%
Health	91.73%	95.04%	95.71%	34.35%	Pet Supplies	91.02%	94.03%	95.02%	38.96%
Overall	Small 3: 91.79%, Large: 94.43%, DS: **95.64%**				Query	Small 3: 0%, Large: 100%, DS: **31.18%**			

Data Shunt for Vision Modality

	Small	Large	DS
Head	70.25%	60.00%	**71.99%**
Med	46.61%	57.28%	**59.91%**
Tail	29.28%	57.19%	**57.61%**
Overall Accuracy	48.84%	58.18%	**63.25%**
Query Proportion	**0%**	100%	66.10%

Evaluation Metrics

- Accuracy: Image classification accuracy
- Query: Sample proportion processed by CLIP

Three Regions

- Head Region: Categories with a large number of samples
- Medium Region: Categories with a moderate number of samples
- Tail Region: Categories with very few samples

Compared to the large model, **Overall accuracy**: 5.07% ↑, **Cost**: ≈33%↓

Data Shunt for Multimodality

	Small	Large	DS
BLEU-1	72.92	73.27	**74.95**
BLEU-2	55.73	60.04	**60.43**
BLEU-3	41.20	**46.99**	46.85
BLEU-4	30.28	**36.11**	35.82
Mean	50.03	54.10	**54.52**
Query Proportion	**0%**	100%	65.36%

Evaluation Metrics

- N-gram BLEU: Quality of machine-generated text
- N-gram: Continuous sequence of *n* items
- Unigram (1-gram): The | Bigram (2-gram): The cat | Trigram (3-gram): The cat sits
- Query: Sample proportion processed by BLIP-2 (1.1B)

Improve in the average BLEU score, while solely **65.36%** of the data is computed by the large model

Ablation for PP and 2CD

	DS	DS-2CD	DS-PP-2CD
Head	**71.99%**	71.21%	71.54%
Med	**59.91%**	58.69%	59.76%
Tail	**57.61%**	56.17%	53.31%
Overall Accuracy	**63.25%**	62.11%	61.63%
Query Proportion	**66.10%**	67.48%	67.48%

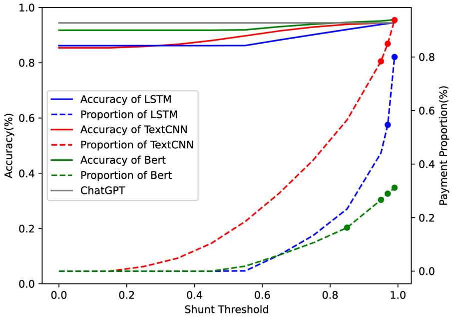
Comparison Algorithm

- DS-2CD: DS without 2-Stage Confidence Distillation
- DS-PP-2CD: DS without Prompt Pruning and 2-Stage Confidence Distillation

Insight

- Both PP and 2CD have a positive impact on the proposed method
- 2CD further reduces the number of large model calls, as small models learn more data distributions
- PP primarily benefits tail data by reducing candidate classes, aligning with the idea that small models assist large models through prior knowledge

Hyperparameter Analysis



- Solid line: Accuracy of DS
- Dotted line: Proportion of query
- Bold dot: DS achieves better performance

Shunt Threshold

- The confidence of a sample is larger than $\delta \rightarrow$ Solely be processed by small models
- Otherwise, processed by large models

Insight

- DS with three different small models (TextCNN, LSTM, fine-tuned BERT) can all surpass the large model
- Better-performing small models allow a wider range for δ

1. Introduction
2. Methodology
3. Experiments
4. Conclusion

Conclusion

Proposed Solution

- *Data Shunt*: Collaborative paradigm for large and small models
- Input is processed by small models first, then passed to large models based on confidence levels
- *Prompt Pruning (PP)*: Small models refine the prediction space for large models
- *2-Stage Confidence Distillation (2CD)*: Large models help small models learn unfamiliar distributions

Experimental Result

Improves performance and reduces the frequency of querying large models across diverse modalities and tasks

Thanks!

Q&A