Determine-Then-Ensemble: Necessity of Top-K Union for Large Language Model Ensembling

Yuxuan Yao, Han Wu† , Mingyang Liu, Sichun Luo, Xiongwei Han, Jie Liu,
Zhijiang Guo, Linqi Song†

香港城市大學 City University of Hong Kong · Noah's Ark Lab · HKUST(GZ)

ICLR

Paper Code

Motivation

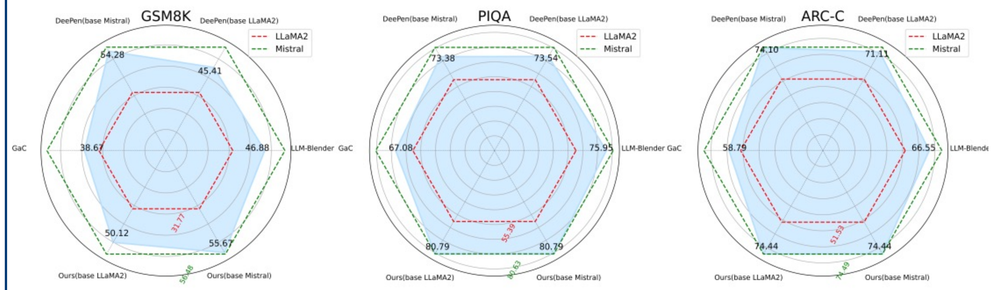
Existing model ensembling methods can be broadly categorized into three types: **output-level**, **probability-level**, and **training-level** approaches. Output-level methods aggregate the complete outputs of multiple candidate models. Probability-level methods, integrate outputs based on probability distributions at each generation step through the intersection or union of the vocabulary. Training-level methods utilize output probability vectors as labels for richer information extraction during training. While output-level methods are constrained by the limitations of existing outputs, and training-level methods introduce additional computational overhead, probability-level methods have garnered particular attention.

Challenges

- First, These approaches concentrate solely on the ensembling technique, sidestepping the crucial discussion of **which types of models can be effectively combined.**
- Second, these methods tend to align the probabilities across the **entire vocabulary at each generation step**. Such a strategy introduces substantial computational overhead during inference, which hinders performance and efficiency

Key Observation

We explored various factors that might affect the performance of model ensembling, including model size (e.g.,3B/7B/8B/13B/70B), model architecture (dense/sparse), performance discrepancies, tokenization strategies (BPE/WordPiece), vocabulary size (e.g., 102K/64K/32K) and tasks variations (e.g., text generation/QA/multiple choices). Finally, we identified three representative factors for further analyses, including **performance discrepancy**, **vocabulary size**, and **tasks variations**.



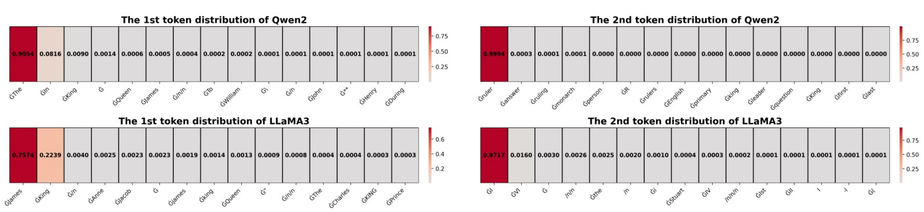
TAKEAWAY I:

Smaller performance gaps lead to greater gains from model ensembling.

Methods	Dataset		Methods	Dataset		Methods	Dataset	
	GSM	PIQA		MMLU	ARC-C		NQ	ARC-C
DeepSeek-7B(102K)	59.67	72.66	DeepSeek-7B(102K)	46.97	58.73	Mistral-7B(32K)	24.25	74.49
Mistral-7B(32K)	56.48	80.63	LLaMA2-13B(32K)	49.61	51.53	Yi-6B(64K)	22.55	73.21
LLM-BLENDER	61.16 (+1.49)	76.54 (-4.09)	LLM-BLENDER	48.86 (-0.75)	57.84 (-0.89)	LLM-BLENDER	22.97 (-1.28)	72.48 (-2.01)
DEEPEN(DeepSeek-7B)	55.00 (-4.67)	77.65 (+4.99)	DEEPEN(DeepSeek-7B)	52.81 (+5.84)	60.00 (+1.27)	DEEPEN(Mistral-7B)	25.26 (+1.01)	76.50 (+1.01)
DEEPEN(Mistral-7B)	61.28 (+4.80)	76.32 (-4.31)	DEEPEN(LLaMA2-13B)	54.09 (+4.48)	62.39 (+10.86)	DEEPEN(Yi-6B)	22.80 (+0.25)	77.86 (+4.65)
Ours(DeepSeek-7B)	62.77 (+3.10)	81.81 (+9.15)	Ours(DeepSeek-7B)	48.90 (+1.93)	60.16 (+1.43)	Ours(Mistral-7B)	24.76 (+0.51)	76.54 (+2.05)
Ours(Mistral-7B)	58.88 (+2.40)	81.81 (+1.18)	Ours(Mistral-7B)	48.90 (-0.71)	60.16 (+8.63)	Ours(Yi-6B)	23.28 (+0.73)	76.54 (+3.33)

TAKEAWAY II:

The influence of vocabulary size for model ensembling is marginal.



Question	who wrote he ain't heavy he's my brother lyrics who was the ruler of england in 1616
Gold answer	Bob Russell James I
LLaMA3 response	Bob Russell ✓ James I ✓
Qwen2 response	He Ain't Heavy, He's My Brother was written by Mike D'Abo. ✗ The song was originally recorded by the British singer-songwriter and released as a single in 1969. It became a hit and has since been covered by numerous artists. . . . The ruler of England in 1616 was King James I. ✓ He was the first monarch of the Stuart dynasty and reigned from March 24, 1603, until his death on July 19, 1625. . . .

TAKEAWAY III:

Even though the performance and vocabulary size are aligned across models, substantial differences in response style could also hinder successful ensembling.

Methodology

Algorithm 1 Union top-k ensembling

Require: LLM$_i$, Vocabulary V$_i$, Tokenizer T$_i$, Demonstration *prompt*, Stop condition *stop*(*)

while not *stop*(*) **do** ▷ Stopping criteria
  TopK$_i$, $P^i \leftarrow$ LLM$_i$(*prompt*) ▷ Generate top-k tokens and their probabilities
  for each model **do**
    if token $w \in V^u$ and $w \in$ TopK$_i$ **then**
      $P^i_{\{w\}}$ and TopK$_i$ remains unchanged.
    else if token $w \in V^u$ and $w \in V^i$ and $w \notin$ TopK$_i$ **then**
      TopK$_i \leftarrow w$, $\hat{P}^i \leftarrow p^i_w$
    else if token $w \in V^u$ and $w \notin V^i$ **then**
      $w^1, \ldots, w^m \leftarrow T^i(w)$
      TopK$_i \leftarrow w^1$, $\hat{P}^i \leftarrow p^i_{w^1}$
    end if
  end for
  $\hat{P}^i_{norm} = \text{softmax}(\hat{P}^i)$, $\hat{P}_{avg} = \frac{1}{n}\sum_{i=1}^{n} \hat{P}^i_{norm}$
  $w = \underset{w \in TopK_i}{\text{argmax}}\,(\hat{P}_{avg})$ ▷ Predict the next token
  prompt $\leftarrow$ *prompt* $+ w$ ▷ Update input sequence
end while

Experiments

(1)UNITE enhances individual model performance when the base models exhibit similar performance levels.

(2)UNITE demonstrates greater robustness and generality.

(3)Collaborating with comparable LLMs does not necessarily yield better results

Method	Dataset				Avg.
	GSM8K	PIQA	ARC-C	MMLU	
LLaMA3	78.77	79.08	79.01	64.58	75.36
LLaMA3.1	80.83	82.86	79.49	66.69	77.47
Qwen2	80.78	84.57	84.92	64.96	78.81
Two-model ensembling (LLaMA3+Qwen2)					
LLM-BLENDER	82.69 (+1.91)	82.53 (-2.04)	82.98 (-1.94)	62.07 (-2.89)	77.57 (-1.24)
DEEPEN	- OOM -				
GAC	80.67 (-0.11)	80.96 (-3.61)	84.93 (+0.01)	67.05 (+2.09)	78.40 (-0.41)
UNITE	84.17 (+3.39)	85.53 (+0.96)	85.07 (+0.15)	69.78 (+4.82)	81.14 (+2.33)
Three-model ensembling					
LLM-BLENDER	83.30(+2.52)	83.47(-1.10)	83.48(-1.44)	62.55(-2.41)	78.20(-0.61)
DEEPEN	- OOM -				
UNITE	84.99 (+4.21)	84.98 (+0.41)	85.39 (+0.47)	69.12 (+4.16)	81.12 (+2.31)