Ensemble Learning for Heterogeneous Large Language Models with Deep Parallel Collaboration

Yichong Huang[†], **Xiaocheng Feng**[†‡✉], **Baohang Li**[†], **Yang Xiang**[‡], **Hui Wang**[‡]
Ting Liu[†], **Bing Qin**[†‡]
[†]Harbin Institute of Technology [‡] Peng Cheng Laboratory
{ychuang,xcfeng,baohangli,tliu,qinb}@ir.hit.edu.cn
{xiangy,wangh06}@ir.hit.edu.cn

Abstract

Large language models (LLMs) exhibit complementary strengths in various tasks, motivating the research of LLM ensembling. However, existing work focuses on training an extra reward model or fusion model to select or combine all candidate answers, posing a great challenge to the generalization on unseen data distributions. Besides, prior methods use textual responses as communication media, ignoring the valuable information in the internal representations. In this work, we propose a training-free ensemble framework DEEPEN, fusing the informative probability distributions yielded by different LLMs at each decoding step. Unfortunately, the vocabulary discrepancy between heterogeneous LLMs directly makes averaging the distributions unfeasible due to the token misalignment. To address this challenge, DEEPEN maps the probability distribution of each model from its own probability space to a universal *relative space* based on the relative representation theory, and performs aggregation. Next, we devise a search-based inverse transformation to transform the aggregated result back to the probability space of one of the ensembling LLMs (main model), in order to determine the next token. We conduct extensive experiments on ensembles of different number of LLMs, ensembles of LLMs with different architectures, and ensembles between the LLM and the specialist model. Experimental results show that (i) DEEPEN achieves consistent improvements across six benchmarks covering subject examination, reasoning, and knowledge, (ii) a well-performing specialist model can benefit from a less effective LLM through distribution fusion, and (iii) DEEPEN has complementary strengths with other ensemble methods such as voting[1].

1 Introduction

With the scaling of model capacities and data volumes, generative large language models (LLMs) have shown impressive language understanding and generation abilities, shedding light for artificial general intelligence [35, 22, 13, 28]. Due to diversities of data sources, model architectures, and training recipes, LLMs have different strengths and weaknesses in various tasks and cases. Therefore, recent research has explored the ensemble of LLMs to exploit the complementary potential [15, 19].

Existing methods can be categorized into selection-based and fusion-based ensembling. Selection-based ensembling selects the best candidate answer from all individual LLMs' answers using an additionally trained reward model [15, 31, 25, 19]. Fusion-based ensembling combines all candidate answers using a trained fusion model [15]. However, these approaches inevitably face significant

[1] Our code is available at: `https://github.com/OrangeInSouth/DeePEn`
✉ means corresponding author.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

challenges in generalizing to unseen data distributions and base models. Besides, prior methods enable collaboration via conveying the textual responses between LLMs while ignoring the rich information (*e.g.,* confidence and alternative answers) in the internal representations.

An ideal solution to this issue is to apply the well-established technology of prediction fusion. [36, 24, 7, 10]. For LLM ensemble, prediction fusion works at each decoding step, averaging the probability distributions from different LLMs to determine the next token. It could not only directly apply to the ensemble of any LLMs without extra parameter training, making it more general, but leverages the informative internal representations (*i.e.,* probability distributions) as communication media. Unfortunately, the vocabulary discrepancy between different LLMs makes it unfeasible to average the distributions due to token misalignment.

In this work, we tackle this key challenge by drawing upon the cross-model invariance of relative representation, which represents each token using the embedding similarities of this token to a set of anchor tokens [21]. Specifically, we propose an ensemble framework **DEEPEN** (**Dee**p **P**arallel **En**semble), enabling distribution fusion for heterogeneous LLMs. DEEPEN transforms the probability distribution from the heterogeneous probability space to a homogeneous relative space, using a matrix formed by the relative representation of all tokens. Next, DEEPEN aggregates the relative representations of all probability distributions in the relative space, coordinating the decision on the next token. Finally, the result of aggregation is transformed back to the probability space of the main model using a search-based inverse transformation to determine the next token.

We conduct extensive experiments ranging from 2-model to 9-model ensembles, covering ensembles of models with parameters ranging from 6B to 70B, ensembles of dense and sparse models, and the ensemble of LLMs with specialist models. Experimental results on six widely-used benchmarks demonstrate that compared to baselines, DEEPEN achieves consistent improvements across all benchmarks. It is also discovered that DEEPEN has complementary strengths when combined with other ensemble methods.

2 Theoretical Analysis

We first introduce relative representation and then illustrate the theoretical support for our method.

2.1 Relative Representation

Previous study discovers that despite the misalignment between latent spaces of different neural networks, the embedding similarity between samples do not change across models [21, 11, 23]. Specifically, Moschella et al. [21] propose relative representation, which represents each sample $x^{(i)}$ by the embedding similarities to a set of anchor samples $\mathbb{A}$ ($x^{(i)}$ and $\mathbb{A}$ are identically distributed):

$$\mathbf{r}_{x^{(i)}} = (cos(e_{x^{(i)}}, e_{a^{(1)}}), ..., cos(e_{x^{(i)}}, e_{a^{(|\mathbb{A}|)}})), \tag{1}$$

where $e_{(*)}$ denotes the embedding of samples, also is absolute representation.

It is empirically evidenced that relative representations possess cross-model invariance, *i.e.,* the relative representation of the same sample keeps invariant across different models, which lays the theoretical foundation for our work to fuse heterogeneous probability distributions.

2.2 Theoretical Support for DEEPEN

Average probability distribution has been widely evidenced to effectively improve the predictive performance in the filed of image and text [2, 10]. For generative language models, as we understand, the underlying mechanism is to interpolate different output semantics represented by the probability distributions. However, for LLM ensemble, vocabulary discrepancy isolates these output semantics in semantic spaces with different basis vectors, making the interpolation infeasible. To tackle this challenge, we aim to enable the cross-model alignment for output semantics, *i.e.,* find a transformation to map the output semantics into a universal space. To this effect, we propose to represent the output semantics with the convex combination of relative representations of all tokens where the weight is the probability assigned to the token.

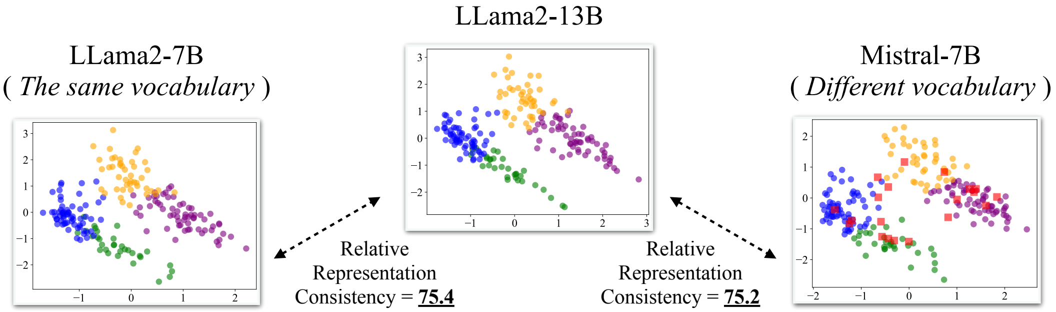


Figure 1: Visualizations for relative representations between models with the same vocabulary and between models with different vocabularies. PCA and K-means clustering are applied only for visualization. Different colors indicate different clusters of samples (word embeddings). The red block indicates the representation of tokens that only appear in Mistral's vocabulary. Relative representation consistency is obtained by calculating the cosine similarity between the relative representations of the same token in different models.

Definition of output semantics in relative space. Formally, given the absolute representation of the output semantics $\mathbf{p}$ and the relative representation matrix $R \in \mathbb{R}^{|V| \times |A|}$ where V is the vocabulary and $A \subseteq V$ is the anchor token set. The i-th row of R is the relative representation of word $w^{(i)}$:

$$R[i] = (cos(e_{w^{(i)}}, e_{a^{(1)}}), ..., cos(e_{w^{(i)}}, e_{a^{(|\mathbb{A}|)}})), \tag{2}$$

and the relative representation of the output semantics $\mathbf{p}$ is defined as: $\mathbf{r} = \mathbf{p} \cdot R$.

Model-invariance of relative representation of output semantic. Next, we illustrate why this representation scheme could align the output semantics isolated in heterogeneous absolute spaces. First, considering two LLMs θ_A and θ_B with the same vocabulary (*e.g.,* LLaMA2-7B and LLaMA2-13B). When expressing the same output semantic, these models output the same probability distribution (*i.e.,* absolute representation) $\mathbf{p}_A$ and $\mathbf{p}_B$. Besides, they have the same (highly similar in practice) relative representation matrix due the vocabulary consistency and cross-model invariance of relative representation. Therefore, the relative representations of output semantics are also identical:

$$\mathbf{r}_A = \mathbf{p}_A \cdot R_A = \mathbf{p}_B \cdot R_B = \mathbf{r}_B. \tag{3}$$

Then, let's consider a language model θ_C with a different vocabulary (*e.g.,* Mistral). Based on the fact that different LLMs typically share mass tokens in their vocabularies (§A), the vocabulary of model θ_C is identical to adding and removing partial tokens to the vocabulary of θ_B, which leads to $\mathbf{p}_B \ncong \mathbf{p}_C$ and $R_B \ncong R_C$. However, in our study, we discover that this change to the vocabulary has not incurred significant influence on the relative representation of the unchanged tokens (*i.e.,* the common tokens between θ_B and θ_C), as shown in Fig. 1. Therefore, we make the reasonable assumption that the local change in the vocabulary could hardly influence the relative space.

3 Methodology

In this section, we first introduce the overall process of our ensemble framework DeepEn and then describe the three parts of DeepEn in detail.

3.1 Overview

We illustrate the process of DeepEn in Fig. 2. Given N models to ensemble, DeepEn first constructs their transformation matrices (*i.e.,* relative representation matrices) mapping the probability distributions from the heterogeneous absolute spaces into the relative space (§3.2). At each decoding step, all models perform prediction and output N probability distributions. These distributions are mapped into the relative space and aggregated (§3.3). Finally, the aggregation result is transformed back into the absolute space of the main model, in order to determine the next token (§3.4).

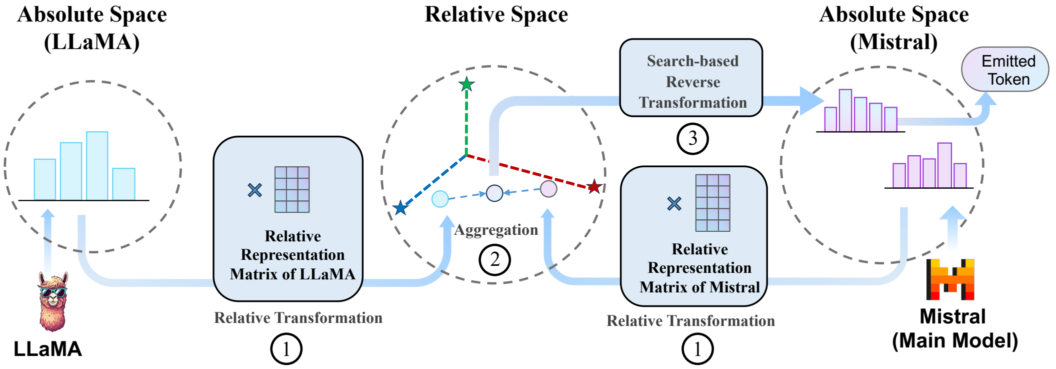


Figure 2: Overview of DEEPEN. The relative representation matrix of each LLM is directly derived by calculating the embedding similarities between each token with the anchor tokens.

3.2 Construction of Relative Transformation

Given N models to ensemble, DEEPEN first finds out the intersection of vocabularies of all models, *i.e.,* common token set C, and samples a subset or uses the full set of common tokens as the anchor token set $A \subseteq C$. Next, for each model, DEEPEN calculates embedding similarities of each token to the anchor words, obtaining the relative representation matrix R (as shown in Eq.2). Finally, to overcome the relative representation degeneration of outlier words, which will be introduced later, we perform normalization on the relative representation of all tokens by a softmax operation so that it becomes a probability distribution. We denote the normalized representation matrix $\hat{R}$:

$$\hat{R}[i] = softmax(R[i]). \tag{4}$$

Anchor Selection. The choice of anchor tokens is crucial for the relative representation capability. Previous research discovers that the capability improves as the number of anchor words increases [21]. Therefore, we employ the full set of common words between LLMs as the anchor words. It is also empirically proved that this method performs more stablely on downstream tasks (§5.2).

Normalization of relative representation matrix. In DEEPEN, the relative representation of each token is normalized by the softmax operation to avoid the relative representation degeneration of outlier words, which are referred to as words that are far away from other words (including the anchors) and become distinguishable in relative space since for being zero vectors. The softmax operation effectively resolves this problem by making each relative representation a probabilistic distribution instead of a zero vector.

3.3 Aggregation in Relative Space

At each decoding step, once each model θ_i outputs the probability distribution $\mathbf{p}_i$, DEEPEN transforms $\mathbf{p}_i$ into the relative representation $\mathbf{r}_i$ using the normalized relative representation matrix: $\mathbf{r}_i = \mathbf{p}_i \cdot \hat{R}_i$, and aggregate all relative representations to obtain the aggregated relative representation:

$$\overline{\mathbf{r}} = \sum_{i=1}^{N} \alpha_i \times \mathbf{r}_i, \tag{5}$$

where α_i is the collaboration weight of model θ_i.

Collaboration Weight. As our work focuses on enabling the distribution fusion of heterogeneous LLMs instead of finding the optimal collaboration weights, we follow the most common practice to uniformly aggregate the distributions ($\alpha = 1/N$, N is the number of models), which is named **DEEPEN-Avg**. Besides, we also adopt a simple and effective method of deducing weights, **DEEPEN-Adapt**, which heuristically sets a larger value to the model with a better performance on the development set: $\alpha_i = s_i / \sum_j s_j$, where $s_i = Acc(\theta_i, \mathcal{D}^{dev}) - \epsilon$, $Acc(\cdot, \cdot)$ indicates the average accuracy of model θ_i on the development set, and ϵ indicates the chance level on the evaluation task. Specifically, $\epsilon = 0$ on the free-form generation tasks and $\epsilon = 1/K$ on the K-choice tasks.

3.4 Inverse Transformation of Relative Representations

To decide the next token according to the aggregated relative representation, DEEPEN aims to transform it from the relative space back to the absolute space of the main model, which is empirically selected with the best-performing model on the development set. To enable this inverse transformation, we adopt a search-based strategy, finding out the absolute representation whose relative representation is identical to the aggregated relative representation. This search problem is formulated as:

$$\overline{\mathbf{p}}_i = \mathop{\arg\min}_{\mathbf{p}_i \in\, \mathbb{P}_i} \ell(\mathbf{p}_i \times \hat{R},\ \overline{\mathbf{r}}), \tag{6}$$

where $\mathbb{P}_i$ denotes the absolute space of model θ_i, and $\ell(\cdot)$ is the loss function to measure the distance between relative representations. In this work, we adopt the KL-divergence due to its convergence.

This search is iteratively conducted under the guidance of the gradient of the loss in Eq.6 with respect to the absolute representation $\mathbf{p}_i$. Specifically, we initialize the start point of searching $\mathbf{p}_i^{(0)}$ with the main model's original absolute representation, and update it as:

$$\mathbf{p}_i^{(t+1)} = \mathbf{p}_i^{(t)} - \eta \times \frac{\partial \ell}{\partial \mathbf{p}_i^{(t)}}, t \in [0, T] \tag{7}$$

where η is an important hyperparameter named the relative ensemble learning rate, and T is the iterations number named relative ensemble learning steps. Finally, we use the updated absolute representation $\mathbf{p}_i^{(T)}$ to determine the emitted token.

4 Experiments

4.1 Experimental Setup

Benchmarks. We mainly conduct experiments on six benchmarks, which can be categorized into:

- **Comprehensive Examination:** (1) MMLU (5-shot) [12], which covers 57 subjects that humans learn, and (2) ARC-C (0-shot) [5], collected from standardized natural science tests.
- **Reasoning Capabilities:** (1) GSM8K [6] (4-shot), which is a dataset of high quality problems at the grade school math level, and (2) PIQA [3] (0-shot), which is a commonsense reasoning dataset.
- **Knowledge Capacities:** (1) TriviaQA (5-shot) [16], collected by Trivia enthusiast authored, and (2) NQ (5-shot) [18], which is a QA corpus consists of queries issued to the Google search engine.

Evaluation. For all benchmarks, we follow the test scripts of OpenCompass leaderboard. Specifically, on the multiple-choice tasks (MMLU, ARC-C, and PIQA), the option with the highest likelihood is selected to calculate the accuracy. On the free-form generation tasks (GSM8K, TriviaQA and NQ), we calculate the exact match (EM) accuracy.

Individual models. As ensemble learning typically works on models with comparable performance [24, 34], we select six well-performing LLMs whose performance are closely matched: LLaMA-2-13B [29], Mistral-7B-v0.1 [13], InternLM-20B [26], Yi-6B [1], Skywork-13B-base [32], and Tigerbot-13b-base-v2 [4]. To achieve better ensemble performance, we conduct experiments on the ensemble of the top-2 models and the top-4 models for each benchmark. Besides, we also consider ensembling various number of models (§4.3) and ensembling more diverse models (§5.1).

Hyperparameters. In this work, we select all of the common tokens between LLMs as the anchor tokens to build the relative spaces, *i.e.,* $A = C$ (§5.2). For the inverse transformation of relative representations, we search the optimal relative learning rate (η in Eq. 7) from 0.05 to 0.30 with an interval of 0.05. We empirically set the number of relative ensemble learning steps $T = 5$ (§5.3).

Comparative methods. We compare DEEPEN with (1) **MINED** [30, 9], which maps the probability distributions of heterogeneous LLMs to the distribution of the main model via aligning tokens in different vocabularies with edit distance, and (2) **LLM-BLENDER** [15], which comprises a reward model PAIRRANKER to score each response of LLMs and a fusion model GENFUSER to fuse

Models	Examination		Reasoning		Knowledge	
	MMLU	ARC-C	GSM8K	PIQA	TriviaQA	NQ
			Individual Models			
LLaMA2-13B	55.07	59.32	29.80	59.68	74.32	28.67
InternLM-20B	59.94	75.81	53.83	64.78	66.88	26.09
Skywork-13B	61.16	66.50	53.90	74.04	58.65	19.75
Tigerbot-13B	51.95	57.44	48.82	68.28	66.22	22.71
Mistral-7B	62.13	73.33	47.50	65.61	73.18	27.62
Yi-6B	63.25	73.33	37.91	76.15	59.02	18.98
			Top-2 Ensemble			
LLM-Blender	63.85 (+0.60)	75.73 (- 0.08)	54.89 (+0.99)	78.31 (+2.16)	74.10 (- 0.22)	28.61 (- 0.06)
MinED	65.04 (+1.79)	77.35 (+1.54)	18.50 (-35.40)	78.98 (+2.83)	72.30 (- 2.02)	28.45 (- 0.22)
DeePEn-Avg	64.68 (+1.43)	77.52 (+1.71)	55.42 (+1.52)	78.87 (+2.72)	75.90 (+1.58)	30.17 (+1.50)
DeePEn-Adapt	65.01 (+1.76)	77.52 (+1.71)	55.65 (+1.75)	79.37 (+3.22)	76.08 (+1.76)	30.69 (+2.02)
			Top-4 Ensemble			
LLM-Blender	61.44 (- 1.81)	71.03 (- 4.78)	43.37(-10.53)	71.16 (- 4.99)	67.87 (- 6.45)	24.18 (- 4.49)
Voting	64.88 (+1.63)	78.41 (+2.60)	63.15 (+9.25)	76.82 (+0.67)	—	—
MBR	—	—	62.09 (+8.26)	—	74.32 (+0.00)	30.28 (+1.61)
MinED	65.61 (+2.36)	78.68 (+2.87)	56.56 (+2.66)	77.87 (+1.72)	71.62 (- 2.70)	29.50 (+0.83)
DeePEn-Avg	65.09 (+1.84)	78.70 (+2.89)	56.18 (+2.28)	77.15 (+1.00)	75.74 (+1.42)	31.55 (+2.88)
DeePEn-Adapt	65.25 (+2.00)	79.15 (+3.34)	56.25 (+2.35)	78.59 (+2.44)	75.76 (+1.44)	31.77 (+3.10)
+Voting/MBR	65.40 (+2.15)	79.44 (+3.63)	65.25 (+11.35)	77.37 (+1.22)	75.65 (+1.33)	32.11 (+3.44)

Table 1: Main results. The best individual model is highlighted in red, and the best ensemble method is highlighted in green, except for the results of the combined method (i.e., the last row). The top-4 models on each benchmark are underlined. '—' indicates that the method does not apply to the task.

candidate responses. In this work, we we only adopt the PairRanker since GenFuser suffers from serious over-generation under our training-free setting. In the ensemble of more than two models, we introduce two additional ensemble methods: (3) **Voting**, which selects the choice favored by most models on the tasks with outputs limited to a fixed set, and (4) **MBR** [8, 17], which selects the answer with the highest textual similarity to other candidate answers. The implementation details of baselines are illustrated in §B.

4.2 Main Results

The main results are shown in Tab. 1, from which we have drawn the following observations:

(1) DeePEn achieves consistent improvements over the individual models. These results prove that our DeePEn successfully enables collaboration between heterogeneous LLMs via aggregating their probability distributions in the relative space. Specifically, DeePEn-Avg achieves improvements of +1.43(MMLU)∼+2.72(PIQA) on the ensemble of top-2 models, and +1.00(PIQA)∼+2.89(ARC-C) on the ensemble of top-4 model. DeePEn-Adapt gains improvements of +1.44(TriviaQA)∼+3.34(ARC-C) on the ensemble of top-4 models.

(2) DeePEn shows better stability than baselines. As shown, LLM-Blender struggles to achieve improvements under the training-free setting. MinED shows unstable performance across different benchmarks. For example, MinED leads to performance drops of -35.40 on the GSM8K benchmark under the top-2 models ensemble setting and -2.70 on the TriviaQA, indicating the limitation of using textual similarity to align tokens in heterogeneous vocabularies. Through case studies, it is revealed that this method of aligning tokens with edit distance disturbs the decoding and produces incomplete words (demonstrated in §7). Instead, DeePEn-Avg achieves consistent improvements and surpasses all baselines in 7/12 settings.

(3) DeePEn has complementary strengths with other ensemble methods. Voting achieves a significant improvement on the mathematical reasoning GSM8K, showing the effectiveness of

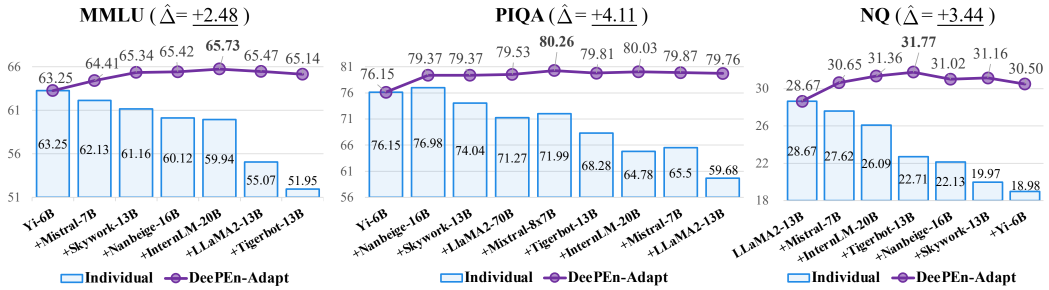


Figure 3: Test set results of ensemble learning on various number of models. Individual models are arranged in descending order of their performance on the development set, and sequentially incorporated into the ensemble. $\hat{\Delta}$ indicates the largest improvement achieved by DEEPEN.

Model	GSM8K	PIQA
LLaMA2-70B	63.84	71.27
Mixtral-8×7B	65.73	71.88
LLM-BLENDER	64.52 (-1.21)	74.54 (+2.66)
MINED	67.10 (+1.37)	75.65 (+3.77)
DEEPEN	67.33 (+1.60)	75.10 (+3.22)

Table 2: Ensemble learning of the *dense* large language model LLaMA2-70B and the *sparse* MoE model Mixtral-8×7B.

Model	En→De	De→En	En→Ro	Ro→En
LLaMA2-13B	30.60	42.27	30.83	39.99
NLLB-600M	32.30	41.49	31.91	42.39
LLM-BLENDER	33.26 (+0.96)	43.28 (+1.01)	33.17 (+1.26)	41.99 ((-0.40))
MINED	27.12 (-5.18)	36.83 (-5.44)	29.91 (-2.00)	34.39 ((-8.00))
DEEPEN	33.34 (+1.04)	43.70 (+1.43)	32.95 (+1.04)	42.84 (+ 0.45)

Table 3: Ensemble learning of the *generalist* model LLaMA2 and the *specialist* translator model NLLB on the translation benchmark Flores-200.

reasoning with multiple paths. To evidence the complementary strength of DEEPEN with VOTING, we combine both methods. On the TriviaQA and NQ, VOTING is replaced with MBR. As shown that the combination of both methods gains a further improvement over VOTING (63.15→65.25).

(4) Collaboration with more worse-performing LLMs is a double-edged sword. The ensemble performance of DEEPEN-Avg with top-4 models surpasses that with top-2 models on 4 benchmarks, but falls short on 2 benchmarks. This is reasonable because incorporating the 3rd and 4th ranked LLMs enhances complementary strengths but also causes the interference with the top-2 models.

4.3 Results on Different Numbers of Models

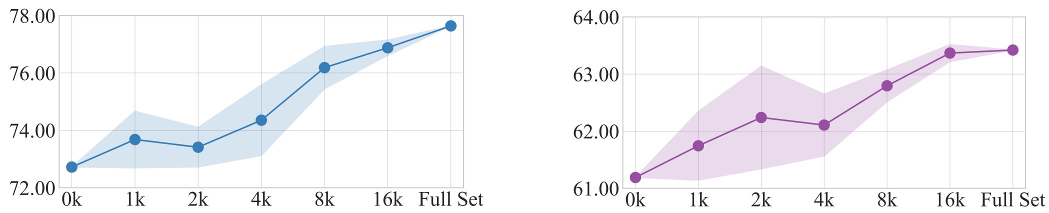
Next, we illustrate the effectiveness of DEEPEN on the ensemble of more models on the MMLU, PIQA, and NQ. We add Nanbeige-16B into the ensemble on all three benchmarks, and add LLaMA2-70B and Mixtral-8×7B on the PIQA due to their comparable performance. As illustrated in Fig. 3, the ensemble performance increases first and then decreases with the joining of more models in descending order of performance. And the ensemble performance peaks in the top-4 or top-5 models across three benchmarks.

5 Analysis

To deeply understand DEEPEN, we first evaluate its performance on the ensemble learning of model sets with diverse architectures, abilities, and performance gaps. Next, we conduct a series of analyses on the reverse transformation process of relative representations.

5.1 Results of Ensembling Diverse Models

Ensemble of the dense model and the sparse model. We first evaluate our method on the ensemble learning of the dense model and the sparse MoE model on the challenge reasoning tasks. Specifically, we use the widely-used large-scale dense model LLaMA2-70B [29] and the popular sparse MoE model Mixtral-8×7B [14] as the base models. As the results shown in Tab. 2, our DEEPEN achieves improvements of +1.60 and +3.22 on the GSM8K and PIQA datasets, even though the base models have achieved a high level of performance.



(a) ARC-C-Dev

(b) MMLU-Dev

Figure 4: Effect of the number of anchor words. The x-axis indicates the number of anchor words randomly sampled from the common words for 4 times.

Methods	**MMLU-Dev**		**TriviaQA-Dev**	
	ACC	Δ	ACC	Δ
Baseline	61.19	–	72.74	–
DeePEn	63.61	+2.42	74.79	+2.05
w/o. **Rel-Norm**	60.73	-0.46	72.95	+0.21

Table 4: Ablation study of normalization on the relative representation matrix to the ensembling performance on the development sets. **Baseline** refers to as the best single model on each benchmark. **DeePEn** refers the performance of ensembling top-2 models in the benchmark.

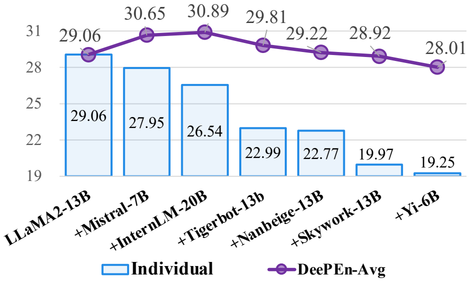


Figure 5: 2-model ensemble of the top-1 model (LLaMA2-13B) with different models on the NQ benchmark, respectively.

Ensemble of the generalist model and the specialist model. To investigate the effectiveness of DeePEn on the ensemble of the generalist model and the specialist model for the specific task, we conduct experiments on the machine translation task using the ensemble of the large language model LLaMA2 and the machine translation model NLLB [27], which is a well-known open-source multilingual translator. We adopt the widely-used machine translation benchmark Flores-200[2]. As the results in Tab. 3 illustrated, DeePEn achieves better translation performance leveraging the diverse translation knowledge in the generalist LLM and the specialist translator.

Ensemble of models with different performance gaps. To assess the stability of DeePEn regarding to the performance gap of base models, we conduct an experiment on the ensemble of model pairs with increasing performance gaps. As the result demonstrated in Tab. 5, the performance of ensemble learning between a well-performing model (the rank-first model)with a worse-performing model could achieve improvements or slightly lag behind the well-performing model.

5.2 Analysis on Relative Transformation

Effect of anchor selection. We demonstrate the impact of different numbers of anchor words through experiments with the top-2 ensemble models on the MMLU and ARC-C datasets. As shown in Fig. 4, an increased number of anchor words can improve performance for LLMs in downstream tasks, and selecting the full set of common words as anchors provides better performance.

Effect of normalization on relative representation matrix. To demonstrate the importance of normalization on the relative representation matrix to the ensemble performance (§3.2), we conduct an ablation analysis. The result is shown in Tab. 4, the ensemble struggles to achieve improvements due to the ineffective representation of outlier words, *i.e.,* words distant to other words. The proportion of outlier words can be derived from the distribution of distance to nearest neighbor words, which is illustrated in Fig. 8. As illustrated, a remarkable proportion ($> 30\%$) of words are distant from other words, *i.e.,* cosine similarity to its nearest neighbor word is less than 0.3. Through the normalization operation, the output semantics that intend to emit outlier words could be prevented from becoming zero vectors by relative transformation.

[2] `https://github.com/facebookresearch/flores`

5.3 Analysis of Reverse Transformation

To better understand the reverse transformation process (§3.4) transforming the relative representation back to the absolute space of the main model, we further analyze each component of this process.

Analysis of relative ensemble learning rates. As shown in Tab. 5, the performance of DeePEn is sensitive to the value of relative ensemble learning rate (η), which is abbreviated by RELR. This observation motivates us to measure the generality of this hyperparameter. Specifically, we illustrate the cross-distribution performance of the searched optimal value of η in Tab. 9. As observed, the optimal value of RELR varies across different datasets, which suggests that the inverse transformation from relative space to absolute space requires adaptive mapping schemes.

RELR (η)	**0.05**	**0.10**	**0.15**	**0.20**	**0.25**	**0.30**
MMLU	+2.42	+1.57	+1.77	+1.96	+1.31	+1.31
TriviaQA	+1.31	+2.05	+1.63	+1.94	+1.82	+1.26

Table 5: Sensitivity analysis of relative ensemble learning rate (**RELR**). We report the improvements of ensembling top-2 models over the best individual models.

Effect of iteration steps in relative ensemble learning. To give a deep view of the dynamics of the inverse transformation in DeePEn, we report the performance change along with different numbers of relative ensemble learning steps (T). Besides, the dynamics of loss of relative ensemble learning (η in Eq. 6)is also reported. As shown in Fig. 9, on the one hand, more steps of relative ensemble learning significantly lead to lower losses. However, the loss is hard to reach zero, *i.e.,* underfitting. On the other hand, increasing the number of steps of relative ensemble learning will cause the performance to increase first and then decrease. The reason behind the performance drop could be that in the early stage of optimization, the focus of optimization is on updating the high-probability tokens. In the later stage of optimization, since the probabilities of all words will be adjusted equally, the high-probability tokens will be interfered with the high-probability ones, thus affecting the performance. Therefore, it is recommended to set a modest value of step number (*e.g.,* $T = 5$).

6 Related Work

Selection-based ensemble. *Rerank* is an intuitive solution to utilize multi-model strengths. Jiang et al. [15] take the first step towards LLM ensemble, training a reward model PairRanker for pairwise comparison on candidate outputs. To overcome the huge computation costs of multi-LLM inference, several works have explored to train a *router* to predict the best-performing model out of a fixed set of LLMs for the given input [31, 25, 19].

Fusion-based ensemble. Towards a synergy between LLMs, Jiang et al. [15] propose GenFuser, trained to combine multiple candidate answers. Different from these training-dependent ensemble methods which pose a great challenge to the generalizability of the reward model or fusion model, our DeePEn is completely training-free, making it more general. Similar to our method, MinED also aims to tackle the vocabulary discrepancy via aligning the tokens in different vocabularies based on edit distance [30, 9]. Unfortunately, this textual similarity-based method exhibits unstable performance and produces abnormal text for LLM ensemble (Tab. 7).

There are several contemporaneous works related to our work. Xu et al. [33] propose EVA to tackle vocabulary discrepancy by learning token alignment between different vocabularies with the assistance of overlapping tokens. Our DeePEn eliminates this training process via directly aligning tokens with the relative representation (more discussion is illustrated in §B). Mavromatis et al. [20] explore adaptive collaboration weights at test time by harnessing the perplexity on the input prompt. We emphasize that this work is complementary to our work.

7 Conclusion

In this work, we propose a training-free LLM ensembling framework DeePEn, which addresses the vocabulary discrepancy when fusing the probability distributions of heterogeneous LLMs. Experimental results on six widely-used benchmarks demonstrate that DeePEn exhibits more stable performance than baseline methods and has complementary strengths with other ensemble methods such as VOTING. We believe our work can inspire further research on the LLMs collaboration, model reuse, and knowledge distillation. In the future, we aim to explore more effective adaptive collaboration schemes to leverage the complementary strengths between different LLMs.

Acknowledgements

Xiaocheng Feng is the corresponding author of this work. We thank the anonymous reviewers for their insightful comments. This work was supported by the National Natural Science Foundation of China (NSFC) (U22B2059, grant62276078), the Key R&D Program of Heilongjiang via grant 2022ZX01A32, the International Cooperation Project of PCL, PCL2022D01 and the Fundamental Research Funds for the Central Universities (Grant No.HIT.OCEF.2023018).

References

[1] . AI. Yi: Open foundation models by 01.ai, 2024. 4.1

[2] Z. Allen-Zhu and Y. Li. Towards understanding ensemble, knowledge distillation and self-distillation in deep learning, 2020. URL https://arxiv.org/abs/2012.09816. 2.2

[3] Y. Bisk, R. Zellers, R. Le bras, J. Gao, and Y. Choi. Piqa: Reasoning about physical commonsense in natural language. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7432–7439, Apr. 2020. doi: 10.1609/aaai.v34i05.6239. URL https://ojs.aaai.org/index.php/AAAI/article/view/6239. 4.1

[4] Y. Chen, W. Cai, L. Wu, X. Li, Z. Xin, and C. Fu. Tigerbot: An open multilingual multitask llm, 2023. 4.1

[5] P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018. 4.1

[6] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, and J. Schulman. Training verifiers to solve math word problems, 2021. 4.1

[7] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma. A survey on ensemble learning. *Frontiers of Computer Science*, 14:241–258, 2020. 1

[8] M. Freitag, B. Ghorbani, and P. Fernandes. Epsilon sampling rocks: Investigating sampling strategies for minimum Bayes risk decoding for machine translation. In H. Bouamor, J. Pino, and K. Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9198–9209, Singapore, Dec. 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.617. URL https://aclanthology.org/2023.findings-emnlp.617. 4.1, B

[9] Y. Fu, H. Peng, L. Ou, A. Sabharwal, and T. Khot. Specializing smaller language models towards multi-step reasoning. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 10421–10430. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/fu23d.html. 4.1, 6

[10] E. Garmash and C. Monz. Ensemble learning for multi-source neural machine translation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1409–1418, Osaka, Japan, Dec. 2016. The COLING 2016 Organizing Committee. URL https://aclanthology.org/C16-1133. 1, 2.2

[11] B. He and M. Ozay. Feature kernel distillation. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=tBIQEvApZK5. 2.1

[12] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ. 4.1

[13] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed. Mistral 7b, 2023. 1, 4.1

[14] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, and et al. Mixtral of experts, 2024. 5.1

[15] D. Jiang, X. Ren, and B. Y. Lin. LLM-blender: Ensembling large language models with pairwise ranking and generative fusion. In A. Rogers, J. Boyd-Graber, and N. Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14165–14178, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.792. URL https://aclanthology.org/2023.acl-long.792. 1, 4.1, 6, 6, B

[16] M. Joshi, E. Choi, D. Weld, and L. Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL https://aclanthology.org/P17-1147. 4.1

[17] S. Kumar and W. Byrne. Minimum Bayes-risk decoding for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 169–176, Boston, Massachusetts, USA, May 2 - May 7 2004. Association for Computational Linguistics. URL https://aclanthology.org/N04-1022. 4.1, B

[18] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee, K. Toutanova, L. Jones, M. Kelcey, M.-W. Chang, A. M. Dai, J. Uszkoreit, Q. Le, and S. Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026. 4.1

[19] K. Lu, H. Yuan, R. Lin, J. Lin, Z. Yuan, C. Zhou, and J. Zhou. Routing to the expert: Efficient reward-guided ensemble of large language models, 2023. 1, 6

[20] C. Mavromatis, P. Karypis, and G. Karypis. Pack of llms: Model fusion at test-time via perplexity optimization, 2024. 6

[21] L. Moschella, V. Maiorca, M. Fumero, A. Norelli, F. Locatello, and E. Rodolà. Relative representations enable zero-shot latent space communication. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=SrC-nwieGJ. 1, 2.1, 3.2

[22] OpenAI. Gpt-4 technical report, 2023. 1

[23] W. Park, D. Kim, Y. Lu, and M. Cho. Relational knowledge distillation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3967–3976, 2019. 2.1

[24] O. Sagi and L. Rokach. Ensemble learning: A survey. *WIREs Data Mining and Knowledge Discovery*, 8(4):e1249, 2018. doi: https://doi.org/10.1002/widm.1249. URL https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/widm.1249. 1, 4.1

[25] T. Shnitzer, A. Ou, M. Silva, K. Soule, Y. Sun, J. Solomon, N. Thompson, and M. Yurochkin. Large language model routing with benchmark datasets, 2024. URL https://openreview.net/forum?id=LyNsMNNLjY. 1, 6

[26] I. Team. Internlm: A multilingual language model with progressively enhanced capabilities. https://github.com/InternLM/InternLM-techreport, 2023. 4.1

[27] N. Team. No language left behind: Scaling human-centered machine translation, 2022. URL https://arxiv.org/abs/2207.04672. 5.1

[28] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample. Llama: Open and efficient foundation language models, 2023. 1

[29] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, and et al. Llama 2: Open foundation and fine-tuned chat models, 2023. 4.1, 5.1

[30] F. Wan, X. Huang, D. Cai, X. Quan, W. Bi, and S. Shi. Knowledge fusion of large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=jiDsk12qcz. 4.1, 6

[31] H. Wang, F. M. Polo, Y. Sun, S. Kundu, E. Xing, and M. Yurochkin. Fusing models with complementary expertise. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=PhMrGCMIRL. 1, 6

[32] T. Wei, L. Zhao, L. Zhang, B. Zhu, L. Wang, and at el. Skywork: A more open bilingual foundation model, 2023. 4.1

[33] Y. Xu, J. Lu, and J. Zhang. Bridging the gap between different vocabularies for llm ensemble, 2024. 6, B

[34] C. Zhang and Y. Ma. *Ensemble Machine Learning: Methods and Applications*. Springer Publishing Company, Incorporated, 2012. ISBN 1441993258. 4.1

[35] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, Y. Du, C. Yang, Y. Chen, Z. Chen, J. Jiang, R. Ren, Y. Li, X. Tang, Z. Liu, P. Liu, J.-Y. Nie, and J.-R. Wen. A survey of large language models, 2023. 1

[36] Z.-H. Zhou. *Ensemble methods: foundations and algorithms*. CRC press, 2012. 1

	Mistral-7B	InternLM-20B	Skywork-13B	LLaMA2-13B	Yi-6B	Tigerbot-13B
Mistral-7B	32,000	26,759	24,983	24,184	24,360	25,121
InternLM-20B	26,759	103,168	41,204	22,566	50,362	44,885
Skywork-13B	24,983	41,204	65,519	32,000	33,646	49,693
LLaMA2-13B	24,184	22,566	32,000	32,000	20,301	32,000
Yi-6B	24,360	50,362	33,646	20,301	64,000	39,360
Tigerbot-13B	25,121	44,885	49,693	32,000	39,360	60,515

Figure 6: Statistics of common words across different vocabularies.

A Statistics of Common Tokens across different LLMs

We count the number of common tokens shared among different LLM vocabularies and present the results in Fig. 6. It is observed that a large number of common words (>20k) exist across the different vocabularies. We also count the number of common tokens in all six LLMs and find that there are a total of 18k common tokens, enabling DEEPEN to be applied to the ensemble learning of a large number of models.

B Details of Baselines

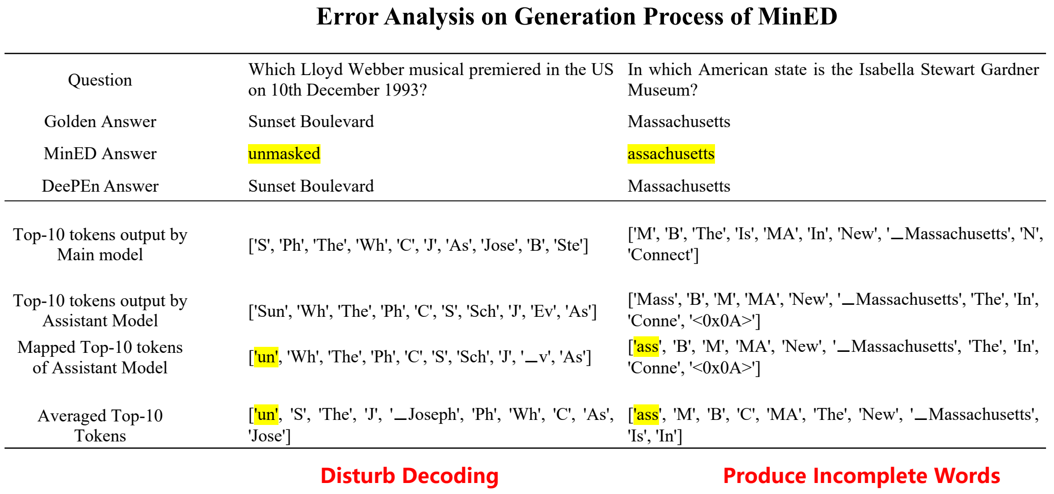
LLM-BLENDER. (1) the selection-based ensemble method **PAIRRANKER** Jiang et al. [15], which is a reward model to score each response of LLMs and (2) the fusion-based ensemble method **GENFUSER** Jiang et al. [15], which is a generative model to fuse multiple candidate responses. Both models are trained on the constructed instruction tuning dataset MixInstruct. In our experiments, as GENFUSER struggles to generate responses following the expected format, we only adopt PAIRRANKER.

VOTING. For tasks with outputs limited to a fixed set (*i.e.,* MMLU, ARC-C, PIQA, GSM8K benchmarks), we adopt the VOTING method on the ensemble learning of more than 2 models. Concretely, we count each candidate answer's occurrences and select the most frequent as the final output. In the event of a tie, the main model's answer is used as the final output.

MBR. For generation tasks, we implement the MBR [8, 17] method, which selects the answer with the highest lexical similarity to other candidate answers. To measure this similarity, we experimented with the edit distance and chrF[3] metrics, ultimately choosing chrF due to its superior performance.

MINED. To bridge the gap between different vocabularies in LLM ensemble, MINED apply the Minimum Edit Distance (MinED) approach to align tokens across different vocabularies, *e.g.,* "get" to "gets". However, this textual similarity-based mapping method could disturb the text generation process and produce incomplete words.

EVA. Recently, Xu et al. [33] propose EVA to tackle the vocabulary discrepancy by learning mappings between the vocabularies of different LLMs with the assistance of overlapping tokens. We have tried to re-implement their method with the released code. However, we encounter a technical problem in that EVA only supports the ensemble learning between LLMs with the same embedding dimension. This is caused by the limitation of tool of vecmap[4], which is used to learn the token alignment.



Error Analysis on Generation Process of MinED

Question	Which Lloyd Webber musical premiered in the US on 10th December 1993?	In which American state is the Isabella Stewart Gardner Museum?
Golden Answer	Sunset Boulevard	Massachusetts
MinED Answer	unmasked	assachusetts
DeePEn Answer	Sunset Boulevard	Massachusetts
Top-10 tokens output by Main model	['S', 'Ph', 'The', 'Wh', 'C', 'J', 'As', 'Jose', 'B', 'Ste']	['M', 'B', 'The', 'Is', 'MA', 'In', 'New', '▁Massachusetts', 'N', 'Connect']
Top-10 tokens output by Assistant Model	['Sun', 'Wh', 'The', 'Ph', 'C', 'S', 'Sch', 'J', 'Ev', 'As']	['Mass', 'B', 'M', 'MA', 'New', '▁Massachusetts', 'The', 'In', 'Conne', '<0x0A>']
Mapped Top-10 tokens of Assistant Model	['un', 'Wh', 'The', 'Ph', 'C', 'S', 'Sch', 'J', '▁v', 'As']	['ass', 'B', 'M', 'MA', 'New', '▁Massachusetts', 'The', 'In', 'Conne', '<0x0A>']
Averaged Top-10 Tokens	['un', 'S', 'The', 'J', '▁Joseph', 'Ph', 'Wh', 'C', 'As', 'Jose']	['ass', 'M', 'B', 'C', 'MA', 'The', 'New', '▁Massachusetts', 'Is', 'In']

Disturb Decoding

Produce Incomplete Words

Figure 7: Analysis of the generation process of MinED. To illustrate the problematic generation process of MinED, we list the top-10 high-probability tokens in the probability distribution of the assistant model and their aligned token.

Models	MMLU-Dev		ARC-C-Dev	
	Indiv	DeePEn	Indiv	DeePEn
Yi-6B	61.19	**63.61** (+2.42)	72.72	**77.55** (+4.83)
Mistral-7B	60.80	**64.46** (+3.66)	73.88	**77.73** (+3.85)

Table 6: Performance of DeePEn with choosing different main models on the development sets. Indiv refers to as individual models. The result of DeePEn indicates the performance of using the model of this row as the main model.

C Additional Experiments

C.1 Choice of main model.

In the process of inverse transformation, DeePEn maps the relative aggregated representation to the absolute space of the main model. Ideally, we expected the results of inverse transformation to keep invariant with the choice of the main model. However, this objective is hard to achieve due to the underfitting observed in the search process. Therefore, we illustrate the performance variance of choosing different main models in Tab. 6. As the results shown on ARC-C, changing the main model from the first-ranked Mistral-7B to the second-rank Yi-6B, the ensemble performance is decreased slightly from 77.73 to 77.55. Interestingly, changing the main model from the rank-1 Yi-6B to the rank-2 Mistral-7B on **MMLU**, the performance is actually improved from 63.63 to 64.46, which indicates that Mistral-7B benefits more than Yi-6B from collaboration. Even so, choosing different main models does not significantly affects the ensemble performance.

C.2 Comparison to Vanilla Prediction Average

To compare our DeePEn with vanilla prediction average, we conduct an experiment for ensembling two LLMs with the same vocabulary and comparable performance on MMLU, *i.e.,* LLaMA2-7B and LLaMA1-13B. As shown in Tab. 7, the performance of DeePEn is comparable, even better than, that of the vanilla prediction average. Theoretically, the performance of the vanilla prediction average is the performance upper-bound of DeePEn. The reason that DeePEn could excel over the vanilla one on MMLU is the under-fitting in the inverse transformation process, which leads to the weights to aggregate the output semantics of different models not being a uniform distribution (*i.e.,* $(0.5, 0.5)$).

[3] https://github.com/mjpost/sacrebleu

[4] https://github.com/artetxem/vecmap

Models	**MMLU-Dev**			**MMLU-Test**		
	INDIV	VANIL	DEEPEN	INDIV	VANIL	DEEPEN
LLaMA1-13B	43.26	45.48	44.37	43.70	45.01	44.22
LLaMA2-7B	42.28		**45.94**	42.99		**45.31**

Table 7: Comparison to vanilla prediction average (VANIL) on the ensemble of LLMs with the same vocabulary.

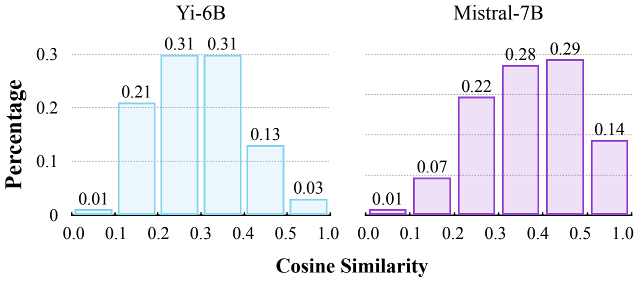


Figure 8: Distance distribution to nearest neighbor words. The distance is measured by calculating the cosine similarity between words.

For example, in Tab. 7, the weights for LLaMA1 and LLaMA2 could be $(0.6, 0.4)$, where the weight of the main model is larger than the other model.

C.3 Latency Analysis

To accomplish the fusion of heterogeneous distributions, DEEPEN first maps the distributions into the relative space and adopts the search-based inverse transformation to map the aggregated relative representation back to the main model's probability distribution, which incurs an extra latency. This latency is mainly caused by the inverse transformation process, which requires T-round search. To demonstrate this latency, we report the token-level inference latency of ensembling two LLMs (Mixtral-8×7b and LLaMA2-70B). This experiment is conducted on 8 A100 GPUs. All of our experiments can be re-implemented on 8 A100 GPUs. As shown in Tab. 8, DEEPEN causes +17% token-level inference latency. However, in practice, this latency could be greatly decreased since all individual models intend to emit the same token in 90% decoding steps. In these steps, we could skip the fusion process and use the consistently agreed token as the next token. In total, DEEPEN actually incurs less than 2% sentence-level inference latency.

	Baseline	$T = 1$	$T = 3$	$T = 5$	$T = 10$
Inference Latency	0.19s	0.20s	0.21s	0.22s	0.24s
Relative Change	0%	+7%	+11%	+17%	+29%

Table 8: Inference Latency of DEEPEN with different search steps T.

D Limitations

As illustrated in Tab. 1, collaboration with more LLMs can sometimes lead to a performance drop caused by interference from lower-performing models. This issue limits the ensemble performance of our current method, even though we have explored setting different collaboration weights for each model on each benchmark (DEEPEN-Adapt). An ideal solution would be to set adaptive collaboration weights at the sample level, or even the token level, for each LLM, which remains a significant

	Baseline	**TrivaQA**	**NQ**	**ARC-C**	**MMLU**
TriviaQA	73.42	75.9	75.41	75.56	75.44
NQ	29.11	30.55	30.65	30.42	30.69
ARC-C	60.29	69.32	72.31	74.19	73.76
MMLU	54.06	59.97	61.04	61.94	61.42

Table 9: Cross-distribution validation of relative ensemble learning rate (η). We report the performance of ensembling LLaMA2-13B and Mistral-7B. Each row indicates the test set used to evaluate performance. Each column indicates the development set used to search the optimal value of η.

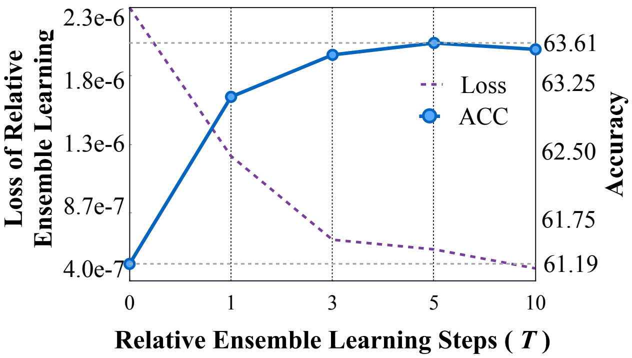


Figure 9: Effect of different number of relative ensemble learning steps.

challenge. Despite this, our work represents an important step towards the distribution fusion of LLMs.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: This paper proposes DEEPEN, a training-free ensemble framework that enables collaboration between large language models with heterogeneous vocabularies by averaging their probability distributions. The key innovation is transforming the probability distributions from each model's vocabulary space into a shared "relative representation" space based on embedding similarities to anchor tokens. The aggregated relative representation is then mapped back to the vocabulary space of one LLM to determine the generated token.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have discussed the limitations of our work in §D.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The method proposed in this work is based on the cross-model invariance of relative representation, which is described in the 'Introduction'.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All the information of our experiments is illustrated in §4.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code will be submitted.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: All the test details are described in §4.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Since our work does not involve with model training, we have not conducted the error statistics.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: As mentioned in §C.3, all of our experiments are conducted on 8 A100 GPUs.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our work conforms the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Our research motivates the research of prediction fusion and model fusion via tackling the vocabulary discrepancy. There is no negative societal impacts of the work.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer:[Yes]

    Justification: All resource used in our work are from open source community.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: This paper does not release new assets

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Not involved

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.