# Weak-to-Strong Honesty Alignment via Learning-to-Rank Supervision

**Yunfan Xie[1], Lixin Zou[1]*, Dan Luo[2], Min Tang[3], Chenliang Li[1]**

[1]Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University,
[2]Lehigh University, Bethlehem, USA, [3]Monash University
{yunfanxie, zoulixin, cllee}@whu.edu.cn, dal417@lehigh.edu, min.tang@monash.edu

## Abstract

Honest alignment refers to the ability of a language model to truthfully convey its knowledge limitations by appropriately refusing to answer questions when it lacks sufficient information. Existing solutions, such as prompt engineering and fine-tuning, face limitations: the former provides only marginal improvements, while the latter struggles to enhance honesty when annotated data is scarce. To overcome the above limitations, we propose WHAT, a novel framework that enhances honesty through weak-to-strong generalization. Specifically, we train the strong LLMs under weak model supervision to improve their honesty. For the weak model, we employ a learning-to-rank strategy to train a "honest head", which learns to select the most honest response among model's outputs generated through beam search. For the strong LLM, we leverage the self-labeled dataset to update its parameters. Our proposal requires only minimal training data to train the weak honest model, yet achieve decent performance for labeling data. In addition, it enables the strong LLMs to have the capabilities to generalize even facing with the flawed label data. Extensive experiments show WHAT significantly boosts honest alignment in large models even with limited labeled data.

## 1 Introduction

In recent years, large language models (LLMs) have achieved significant progress in tasks like code generation (Chen et al., 2021; Guo et al., 2024; OpenAI, a), mathematical reasoning (Guo et al., 2025; OpenAI, b), and scientific research (García-Ferrero et al., 2024; Wang et al., 2023), driven by scaling paradigm (OpenAI, 2023; Kaplan et al., 2020; Touvron et al., 2023; Snell et al., 2024). However, LLMs still frequently produce outputs that are factually inconsistent or lack grounding (Huang et al., 2025; Ji et al., 2024b; Tang et al., 2025), undermining their reliability. To address this, improving "model honesty" – defined as providing accurate responses within known domains and explicitly acknowledging uncertainty beyond model's knowledge scope – has become a critical goal in the research community (Cheng et al., 2024; Li et al., 2024b; Askell et al., 2021).

However, achieving honesty in LLMs remains challenging, as current alignment practices often prioritize helpfulness and harmlessness over honesty (Yang et al., 2024a). This imbalance stems from the difficulty of creating model-specific alignment data, tailored examples that teach models to distinguish between known and unknown domains. Without such data, it is difficult to define precise knowledge boundaries of a model; thus, models risk either overconfidence in uncertain contexts or excessive refusal of valid queries.

Efforts to enhance model honesty can be divided into two main approaches: prompt engineering and model fine-tuning. When provided with well-designed prompts, LLMs can improve its honesty (Brown et al., 2020; Wen et al., 2024). Some studies suggest using carefully crafted prompts to encourage models to be more cautious when answering questions about unknown knowledge (Yang et al., 2024a; Xu et al., 2024). In addition, other research focuses on improving LLM's honesty via fine-tuning. These approaches first determine whether the model possesses knowledge about a given question. When the model lacks sufficient knowledge, it is trained to express uncertainty. This training can be carried out through supervised fine-tuning (Yang et al., 2024a; Wan et al., 2024; Cheng et al., 2024), direct preference optimization (Rafailov et al., 2023; Cheng et al., 2024), or proximal policy optimization (Schulman et al., 2017; Xu et al., 2024) and their variants (Xu et al., 2024).

While these methods improve honesty to some

*Corresponding author.



*Findings of the Association for Computational Linguistics: ACL 2025*, pages 10154–10168

extent, they face significant limitations. On the one hand, prompt engineering yields only modest improvements since the training process does not explicitly align with honesty objectives. On the other hand, fine-tuning methods achieve better results but depend heavily on high-quality, diverse training data, which is often limited, particularly for complex questions where answers may be diverse or subjective. Evaluating such answers is inherently challenging, further complicating data curation. Moreover, each target model architecture may exhibit unique knowledge gaps; thus it requires tailored dataset construction. This customization imposes significant computational and human resource demands, making large-scale fine-tuning less scalable and harder to generalize across varied scenarios.

To address these challenges, we introduce WHAT (**W**eak-to-Strong **H**onesty **A**lignmen**T** via Learning-to-Rank Supervision), a framework inspired by weak-to-strong generalization (Burns et al.). This approach leverages a lightweight "honest head" network to guide LLMs toward more truthful outputs, even under limited supervision. Specifically, the key mechanisms of our proposal consist of three parts: (1) **Honest Head Training**: An honest head is a lightweight model that is trained on available labeled data using a learning-to-rank framework. It is then used to identify the most honest responses among those generated by LLMs through beam search. This design mitigates data scarcity by requiring minimal labeled data and emphasizing comparative honesty over absolute ground truth. **(2) Large-Scale Self-Labeling**: For questions lacking reference answers, we combine the honest head's predicted scores with the LLM's prediction probabilities to generate pseudo label. This mechanism extends honesty alignment to unlabeled data, significantly reducing reliance on human annotations and mitigating data scarcity. **(3) Weak-to-Strong Fine-Tuning**: The synthesized dataset is used to fine-tune the base LLM, merging the honest heads specialized scoring (weak supervisor) with the LLMs generalization power (strong model). This avoids model-specific dependencies by creating a unified training pipeline adaptable to diverse architectures.

Our contributions are summarized as follows:

- We introduce a method that leverages weak supervision to improve the honesty of large language models, reducing the dependency on extensive data annotation.
- We propose a novel approach, WHAT, which enhances the honesty of large language models by effectively ranking their outputs. It requires minimal training cost and introduces nearly zero computational overhead during inference.
- Through extensive experiments, we demonstrate that WHAT significantly enhances the honesty of LLMs compared to existing methods, all while better preserving the models intrinsic knowledge.

## 2 Preliminary

This section presents preliminary knowledge regarding language model decoding and honesty measurement, which is helpful for understanding the subsequent methodology.

### 2.1 Decoding Process in LLM

In the decoding process of a large language model $M$, given an input token sequence $q$, the model can generates token at timestep $t$, where the probability distribution over the vocabulary is:

$$P(y_t \mid y_{<t}, q) = M(y_t \mid y_{<t}, q),$$

where $y_{<t}$ represents the sequence of previously generated tokens. Afterwards, the model employs various sampling strategies to sample $n$ candidates as $\{y_{1,t}, \ldots, y_{n,t}\}$. The final decoding sequence $y^*$ is chosen based on an accumulated score, which is typically the log probability of the entire sequence:

$$Q(y^* \mid q) = \max_{y_{i,t}} \sum_{t=1}^{T} \log P(y_{i,t} \mid y_{i,<t}, q). \quad (1)$$

### 2.2 Measurement of Honesty

The measurement of honesty involves two steps: probing the model's knowledge boundary and assessing its response's honesty.

**Probing the Knowledge Boundary** This step aims to assess whether the question falls within models knowledge domain. Let $q$ be a question, and $a$ be the reference answer to question $q$. We use LLM $M$ to generate a set of responses $\mathcal{Y} = M(q, a)$. The honesty of a response $y_i \in \mathcal{Y}$ is evaluated by the correctness function $J(y, q, a)$ (Yang

et al., 2024a), which is defined as:

$$J(y,q,a) = \begin{cases} 1, & \text{if } y \text{ is the correct answer,} \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Here, $J(y,q,a)$ can be implemented through straightforward term matching or more sophisticated LLM judging (Li et al., 2024a).

To this end, we probe the models knowledge boundary through the lens of its generated responses, *i.e.,* the proportion of correct responses in the set $\mathcal{Y}$ that exceeds a predefined threshold $\alpha$:

$$k(M,q) = \begin{cases} 1, & \text{if } \frac{\sum_{i=1}^{n} J(y_i,q,a)}{|\mathcal{Y}|} > \alpha, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

**Estimating Honesty value** Next, we present the honesty value estimation for LLM-generated responses. We first introduce a category function $\mathcal{C}$, which classifies a response $y$ based on its explicit acknowledgment of uncertainty and its correctness as defined in Eq. 2:

$$c(y,q,a) = \begin{cases} 0, & \text{if the response expresses} \\ & \text{uncertainty in } y, \\ 1, & \text{if } J(y,q,a) = 1, \\ -1, & \text{if } J(y,q,a) = 0. \end{cases} \tag{4}$$

Then, the honesty value $v(y,q,a)$ of the models response to a question is computed as:

$$v(y,q,a) = \begin{cases} 3, & \text{if } c(y,q,a) = 1, \\ 2, & \text{if } c(y,q,a) = 0 \\ & \text{and } k(M,q) = 0, \\ 1, & \text{if } c(y,q,a) = 0 \\ & \text{and } k(M,q) = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

The rationale behind the honesty value function is as follows: the model receives the highest honesty value only when it provides a correct response. Additionally, when it explicitly acknowledges uncertainty, it is awarded a higher score when the question genuinely falls outside its knowledge scope. This design discourages excessive refusal of valid queries. Finally, the model receives a score of zero, strictly penalizing responses that exhibit overconfidence despite uncertainty.

Ideally, to align LLMs for honesty, we can optimize the decoding process by generating responses with highest honesty value. However, in practice, assessing the accuracy of responses generated by LLMs is not always feasible due to the absence of human-annotation or the scale of the model itself, especially when the model is sufficiently large and complex to preclude straightforward evaluation. We propose our solution in the next section.

## 3 Weak-to-Strong Honest Generation

This section describes our method to generate pseudo-labels using a lightweight, easily trainable weak model, thereby reducing dependence on manual data annotation. Figure 1 provides an overview of our approach, WHAT. Our method is composed of three phrase: (1) **Weak Honest Model Training**: We train a lightweight "honest head" model on limited labeled data within a learning-to-rank framework. This model is used to identify the most honest responses among those generated by LLMs through beam search. (2) **Large-scale Self-labeling**: The honest head model generates pseudo-labels for unlabeled responses, which provides reliable supervision for unseen instances. (3) **Weak-to-strong Fine-tuning**: Using the pseudo-labeled data, we fine-tune a stronger, more robust model, enhancing its performance while minimizing manual annotation costs.

### 3.1 Weak Honest Model Training

To effectively guide the strong model during training, we first need a guide model to adapt to the target task. Through empirical observations, we find that although LLM-generated responses may exhibit dishonest behavior, the candidate responses in their beam search outputs often contain some honest alternatives. This inspires us to adopt a reranking strategy to select honest responses from model outputs. We refer to the weak model as the *honest head*. Below, we detail honest head's architecture design, input selection strategy, and learning to rank training approach.

**Architecture Design** We instantiate the honest head as a 3-layer MLP upon LLMs. In particular, this design leverages two key insights: (1) Transformer intermediate layers has encoded rich linguistic features sufficient for downstream tasks (Xie et al., 2025; Meng et al., 2022), and (2) simplified architectures facilitate training while maintaining the effectiveness during inference. Formally, we estimate the honest score as:

$$s(y) = W_3 \cdot \sigma(W_2 \cdot \sigma(W_1 h + b_1) + b_2) + b_3, \tag{6}$$

where $W_i \in \mathbb{R}^{d_i \times d_{i-1}}$ and $b_i \in \mathbb{R}^{d_i}$ denote weight matrices and bias terms, respectively ($d_0$ input dimension, $d_1, d_2$ hidden dimensions, $d_3 = 1$ as the output dimension, and $\sigma$ represents the activation function. In particular, $h \in \mathbb{R}^{d_0}$ represents the hidden representation of the last tokens hidden state in response $y$, extracted from the selected transformer layer. The layer selection strategy is detailed in the following section.

**Intermediate Layer Selection** Selecting appropriate hidden layers for honest head is crucial. We formulate the layer selection as an empirical optimization problem: for a $N$-layer transformer, we evaluate hidden states $\{h_i\}_{i=1}^{N}$ through grid search and select the layers that maximize honest score in Eq. 5 on the validation set. We select the final token for the honesty value computation.

**Optimizing Honest Head via Learning to Rank** Given the multi-level graded dataset for honest, one straightforward approach is to optimize the honest head by mapping from $s(y)$ to annotation score. However, this pointwise learning often suffers from inaccuracies in the mapping process. Our empirical observations demonstrate that beam search outputs frequently contain honest alternatives among the candidate responses. Thereforewe propose to optimize the honest head through a learning-to-rank approach, *i.e.,* selecting the most honest response from $n$ candidate responses for each input query.

Let $\{v_i\}_{i=1}^{n}$ be the *annotated* ground-truth honesty values and $\{s_i\}_{i=1}^{n}$ be the *predicted* honesty values produced by our honest head. In this section, we explore two widely used LTR loss, *i.e.,* the pairwise ranking loss and the listwise ranking loss, which enables the honest head learns to select the most honest response.

We propose a **pairwise LTR** framework. For a pair of responses $oy_i$ and $y_j$, the probability of $y_i$ being more truthful than $y_j$ under our model's predicted scores is defined as:

$$P(y_i \succ y_j) = \frac{\exp(s_i)}{\exp(s_i) + \exp(s_j)}. \quad (7)$$

Then, we minimize the negative log-likelihood of these observed pairwise preferences:

$$\mathcal{L}_{\text{pair}} = -\sum_{i=1}^{n} \sum_{j \neq i} \mathbb{I}_{ij} (v_i \succ v_j) \log P(y_i \succ y_j), \quad (8)$$

where $\mathbb{I}_{ij} = 1$ if the annotated honest score $v_i$ exceeds $v_j$, and 0 otherwise. The honest head is optimized to align pairwise order with the annotated ground-truth of honesty.

Despite its effectiveness, the pairwise ranking can lead to quadratic growth in computational complexity, (*i.e.,* constructing $C_N^2$ pairs), making it inefficient as the number of candidates increases via beam search. To address this limitation, we propose **listwise LTR**. Particularly, it formulates the evaluation of a ranked list as a process of attention allocation (Bruch et al., 2019). The best attention allocation strategy on a list of responses $\{y_i, \ldots, y_n\}$ is defined as:

$$a_i = \frac{\exp(v_i)}{\sum_{j=1}^{n} \exp(v_j)}. \quad (9)$$

Similarly, we compute the attention distribution of our honest head with the ranking score $\{s_i\}_{i=1}^{N}$ and use the cross entropy between our attention strategy and the best attention strategy as the loss:

$$\mathcal{L}_{\text{list}} = -\sum_{i=1}^{n} a_i \log\left(\frac{\exp(s_i)}{\sum_j \exp(s_j)}\right). \quad (10)$$

### 3.2 Large Scale Self-labeling

In this section, we elaborate how to self-label with the honest head. The honest head may inevitably capture biases present in the data, solely depending on honest head may lead to suboptimal. Therefore, we combine honesty values with the language model's intrinsic likelihoods through ensemble decoding to generate pseudo-labels.

Let $\mathcal{Q}_u$ be a set of unlabeled questions. For each question $q \in \mathcal{Q}_u$, language model $M$ will generate $K$ candidate sequences $\{y_j^q\}_{j=1}^{K}$ using beam search algorithm, we first obtain two scores for each candidate: the language model score $Q(y_j^q \mid q)$, defined in Eq. 1 and the predicted honesty value $s(y_j^q)$ defined in Eq. 6. We then combine these two terms as the final score via soft-attention normalization:

$$\begin{aligned} \hat{Q}(y_j^q \mid q) &= \frac{\exp\big(Q(y_j^q \mid q)\big)}{\sum_{j=1}^{n} \exp\big(Q(y_j^q \mid q)\big)}, \\ \hat{s}(y_j^q) &= \frac{\exp\big(s(y_j^q)\big)}{\sum_{j=1}^{K} \exp\big(s(y_j^q)\big)}, \\ z(y_j^q \mid q) &= (1-\beta)\hat{Q}(y_j^q \mid q) + \beta\, \hat{s}(y_j^q), \end{aligned} \quad (11)$$

where $\beta \in (0, 1)$ is a hyperparameter termed "honesty ratio"; it governs the contribution of honest

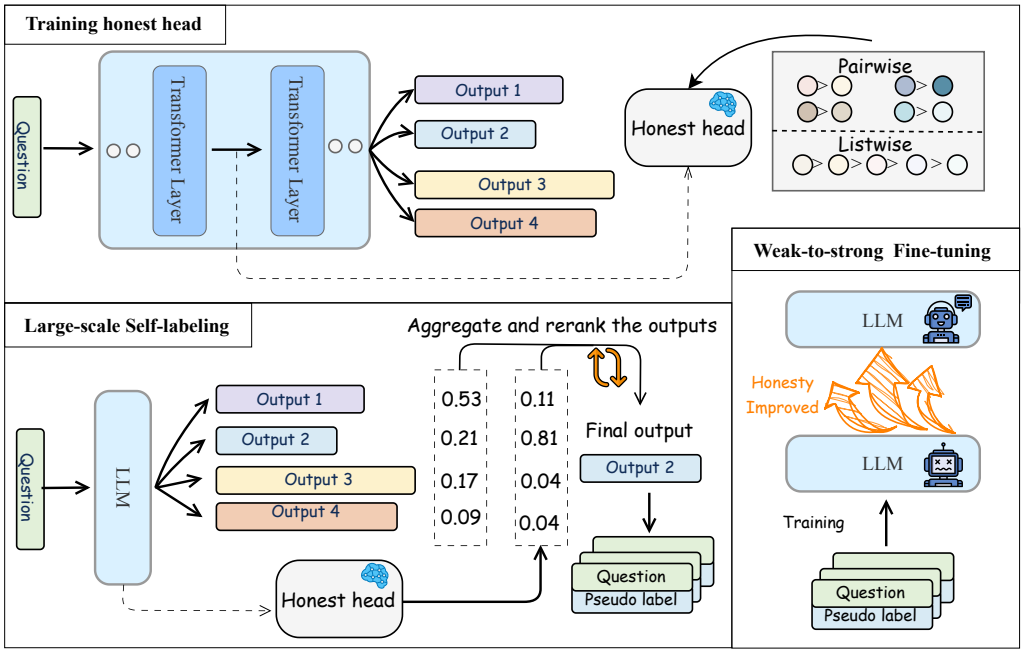


Figure 1: An overview of WHAT. The honest head module takes the hidden states generated by the LLM as input and is trained using an LTR (Learning-to-Rank) loss function. Large Scale Self-labeling: Predicted scores from the honest head and the response probabilities from the LLM are ensembled to self-label the questions. Weak-to-strong Fine-tuning: Using the self-labeled dataset for further model training.

head relative to the language model's intrinsic likelihood. When $\beta = 0$, this reduces to standard Best-of-N sampling. In practice, $\beta$ is tuned via a simple trial-and-error procedure on a validation set.

For each question $q \in \mathcal{Q}_u$, its most honest response is selected among the $n$ generated responses according to the final score $z(y_j^q \mid q)$ as:

$$y_q^* = \arg\max_{y_j^q} \left[ z(y_j^i \mid q^i) \right]. \tag{12}$$

To this end, we can obtain the self-labeled dataset $\mathcal{D}_u = \{q, y_q^\star\}_{q \in \mathcal{Q}_u}$ that consists of unlabeled questions and its corresponding most honest response via pseudo-labels.

### 3.3 Weak-to-Strong Generalization

Through honest head guided decoding, we have already enhanced the model's honesty. To further generalize the model's ability, we fine tune the model $M$ using the the self-labeling dataset $\mathcal{D}_u$ collecting in Section 3.2.

A large languagde model parameterized by $\theta$ are optimized by minimizing the negative log-likelihood loss over the self-labeled dataset:

$$\mathcal{L}(\theta) = -\frac{1}{N} \sum_{q \in \mathcal{Q}_u} \log Q\left(y_q^\star | q; \theta\right), \tag{13}$$

where $Q\left(y_q^\star | q; \theta\right)$ denotes the probability of the pseudo-labeled output $y_q^\star$ given input $x^i$ and model parameters $\theta$. The model parameters $\theta$ are updated to minimize this loss, resulting in the fine-tuned model $M^*$. This supervised fine-tuning process aligns the model's output distribution with the high-quality pseudo-labels, enhancing both honesty and overall performance.

In summary, the pseudo-label are generated by a weak model (*i.e.,* small honest head), which are then used to enhance the honesty of a strong model (*i.e.,* large language model). This weak-to-strong design enables our proposal has high generalization capabilities. In particular, WHAT achieves strong results even with scarce annotations due to two synergistic factors. On the one hand, the honest head uses a lightweight architecture (fewer parameters than the LLM) and a learning-to-rank loss. This allows it to effectively rerank the LLM's outputs by honesty using minimal data. On the other hand, the LLM already possesses robust representations, enabling it to leverage pseudo-labels for further alignment. For unseen data, even imperfect pseudo-labels from the honest head allow the LLM to refine its latent capabilities.

## 4 Experiments

### 4.1 Experimental Setup

**Baselines** To evaluate the effectiveness of our approach, we compared it against four baseline methods. **(1) Prudent Prompt**: This method provides the model with explicit instructions designed to encourage cautious reasoning and knowledge-aware responses. **(2) In-Context Learning (ICL)**: The model is conditioned on four task-specific demonstrations to guide its reasoning. **(3) Supervised Fine-Tuning (SFT)**: Due to computational constraints, we employed the LoRA fine-tuning approach (Hu et al., 2022a), a parameter-efficient method that adapts the model via low-rank updates. **(4) Direct Preference Optimization (DPO)**: This is a reinforcement learning-based optimization framework that aligns model outputs with human preferences. All prompts for these baselines are provided in Appendix B.

**Datasets** We conducted experiments on three datasets: **1) PopQA** (Mallen et al., 2022) is a large-scale, open-domain question answering dataset consisting of entity-centric QA pairs. Each question is created by converting a knowledge tuple retrieved from Wikidata using a template. **2) SQuAD** (Rajpurkar et al., 2016) is a reading comprehension dataset that contains questions posed by crowdworkers based on a set of Wikipedia articles. **3) Non-AmbigQA** is a subset of the NQ-Open dataset (Kwiatkowski et al., 2019), consisting of clear and unambiguous questions along with their corresponding answers.

**Experimental Details** In our experiments, we evaluated three open-source models: LLama3-8b-instruct (Dubey et al., 2024), Gemma2-9b-instruct (Team et al., 2024), and Mistral-7B-Instruct-v0.3. For brevity, we refer to these models as Llama, Gemma, and Mistral, respectively. We set $\alpha$ to 0.1. For supervised fine-tuning, we set the learning rate to 1e-5 and used 2 epochs. Due to resource constraints, we utilized LoRA for fine-tuning (Hu et al., 2022b; Li et al., 2025). During generation, models were configured with a temperature of 0.8 to encourage diversity. The honest head module was trained for up to 40 epochs. We reserved 10% of the training data as a validation set to select the optimal layers hidden state as input to the honest head, based on validation performance. Both our proposed model (WHAT) and the DPO baseline underwent initial task-specific fine-tuning using labeled data to align outputs with task requirements. This fine-tuning used identical hyperparameters to SFT. Subsequently, the fine-tuned models generated outputs across the full training set, which were used to construct the honest heads training data. The DPO baselines training data construction mirrored that of $\text{WHAT}_{\text{Pair}}$. For all datasets, we sampled 2,000 examples from the training set (retaining answers) and removed answers from the remaining data to simulate unannotated questions. For datasets with provided documents, we discarded the documents and fed only the questions into the language model (LLM). Dataset statistics are detailed in Appendix A.

**Evaluation metrics** Following the methodology of Yang et al. (2024a), we evaluated model performance using three metrics: **1) Prudence Score**: Measures the probability that a model expresses uncertainty when encountering unknown questions. **2) Over-Conservativeness Score (Over-Conserv.)**: Quantifies the probability that a model expresses uncertainty when responding to known questions. **3) Honesty Score**: Defined as the arithmetic mean of the Prudence Score and Over-Conservativeness Score, this metric reflects the models overall honesty. To assess the correctness of model responses, we input the question, reference answer, and model-generated reply into DeepSeek-V3 (Liu et al., 2024a) for automated analysis. The specific prompt used for this evaluation is provided in Appendix B.

### 4.2 Main Experimental Results

Table 1 presents the experimental results across three datasets. From the table, we have following key observations: **1) Our method mitigates the challenge of scarce labeled data and achieves state-of-the-art (SOTA) performance.** While fine-tuning improves honesty scores, it underperforms in low-data regimes. For example, on the SQuAD dataset, Gemma achieves a 14% honesty score gain via supervised fine-tuning (SFT) compared to baseline prompt engineering. Our approach further boosts performance by 9.9%. **2) Prompt engineering demonstrates limited effectiveness in aligning with "Honesty".** Despite efforts to enhance honesty via prompt-based or in-context learning (ICL) methods, models exhibit low prudence and over-conservativeness scores, indicating reluctance to express uncertainty. **3) Listwise optimization outperforms**

**pairwise methods for training the honest head.** The listwise approach surpasses pairwise comparison in 7 of 9 experimental settings (3 models $\times$ 3 datasets). This advantage stems from listwise learnings direct optimization of full output rankings, which better aligns with task objectives than pairwise local comparisons.

The further comparisons can be found in Appendix C.

### 4.3 Ablation Study

**Influence of Weak-to-Strong Fine-Tuning** In this subsection, we analyze the impact of weak-to-strong generation using self-labeled data. For comparison, we employ two baseline methods: (1) Supervised Fine-Tuning (SFT) and (2) Honest-aware Decoding, where the trained honest head guides the decoding process based on the methodology outlined in Section 3.2. We conduct experiments on the PopQA dataset. The results presented in Figure 2 reveal two key insights. First, **the honest head significantly improves model performance.** For instance, on the PopQA dataset, Honest-aware Decoding boosts accuracy by 5.2% compared to SFT for the Llama model. Second, **weak-to-strong fine-tuning further enhances performance**: continued training with self-labeled data leads to an additional 2.3% improvement over Honest-aware Decoding for Llama on the same dataset. These findings highlight the importance of leveraging annotated data for incremental model optimization.

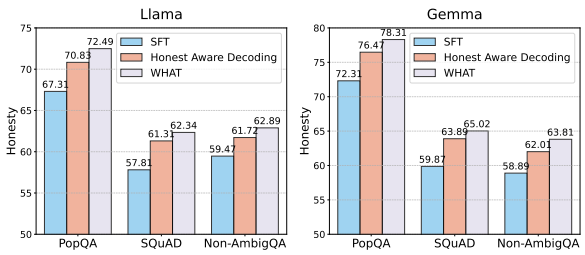


Figure 2: Effectiveness of weak-to-strong fine-tuning.

**Influence of Input Selection Strategy** In this subsection, we examine the impact of using hidden states from different transformer layers to train the honest head module. As depicted in Figure 3, selecting the appropriate hidden state layer as input for the honest head is critical, as different layers yield significantly varying effects on model performance. Notably, despite undergoing extensive processing through transformer layers, **the final layer does not always represent the optimal choice for honest head training.**

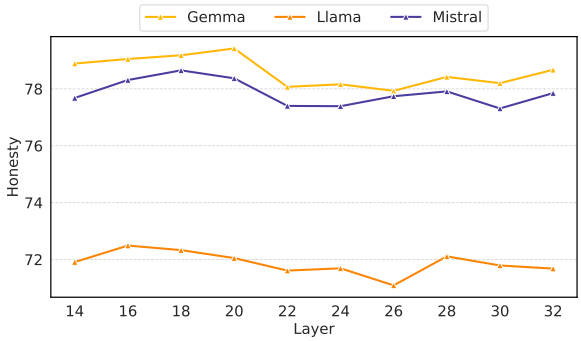


Figure 3: Analyzing the impact of hidden states from various network layers as inputs to the honest head for performance.

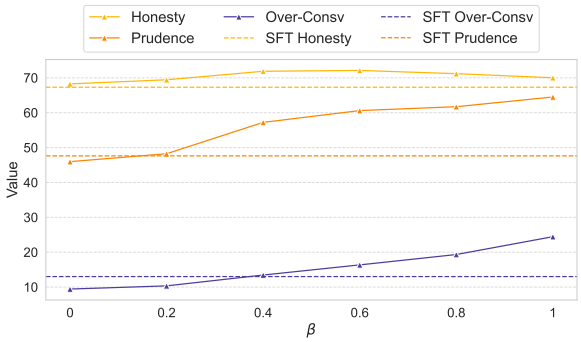


Figure 4: The sensitivity analysis of honesty ratio $\beta$ on the performance on WHAT$_{List}$ applied to Llama on PopQA dataset.

**Influence of Honesty Ratio** In this subsection, we examine the impact of honesty ratio hyperparameter $\beta$. Figure 4 presents the performance of Llama on the PopQA dataset across different honesty ratios. When $\beta$ is set to 0, the Best-of-N method is essentially used to select answers from model responses as pseudo-labels, resulting in only minor improvements in model performance. When $\beta$ is set to 1, decoding is solely guided by pseudo-labels generated by the honest head. Under this configuration, the model's over-conservativeness score rises substantially, leading to a decrease in honest scores. This indicates that the honest head may absorb unintended biases from the training data, emphasizing the need to incorporate the log-likelihood of an ensemble LLM to counteract this effect. Thus, it demonstrates that **selecting appropriate honesty ratio is crucial.**

| | | PopQA | | | SQuAD | | | Non-AmbigQA | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Prudence ↑ | Over-Consv. ↓ | Honesty ↑ | Prudence ↑ | Over-Consv. ↓ | Honesty ↑ | Prudence ↑ | Over-Consv. ↓ | Honesty ↑ |
| Llama | Prompt | 3.73 | 8.13 | 47.80 | 5.59 | **2.74** | 51.43 | 16.78 | **4.85** | 55.97 |
| | ICL | 5.40 | **4.44** | 50.48 | 7.25 | 2.59 | 52.33 | 13.11 | 5.41 | 53.85 |
| | SFT | 47.63 | 13.01 | 67.31 | 29.09 | 13.47 | 57.81 | 32.15 | 13.21 | 59.47 |
| | DPO | 49.31 | 11.35 | 68.98 | **34.70** | 15.88 | 59.41 | 29.10 | 11.52 | 58.79 |
| | WHAT $_{\text{Pair}}$ | 59.31 | 12.63 | **73.34** | 33.14 | 10.56 | 61.29 | **40.09** | 15.86 | 62.12 |
| | WHAT $_{\text{List}}$ | **60.23** | 15.25 | 72.49 | 33.78 | 9.10 | **62.34** | 39.69 | 13.91 | **62.89** |
| Gemma | Prompt | 24.89 | **9.87** | 57.51 | 13.69 | **5.37** | 54.16 | 16.21 | 6.71 | 54.75 |
| | ICL | 22.48 | 10.04 | 56.22 | 15.49 | 6.81 | 54.34 | 19.31 | 5.89 | 56.71 |
| | SFT | 65.32 | 19.94 | 72.69 | 38.10 | 14.58 | 61.76 | 35.81 | 10.37 | 62.72 |
| | DPO | 67.23 | 23.29 | 71.97 | 39.81 | 15.57 | 62.12 | 34.92 | **5.14** | 64.89 |
| | WHAT $_{\text{Pair}}$ | 67.31 | 10.63 | 78.34 | 45.07 | 11.62 | 66.72 | 43.33 | 11.27 | 66.03 |
| | WHAT $_{\text{List}}$ | **69.31** | 10.95 | **79.18** | **46.09** | 10.31 | **67.89** | **45.10** | 10.68 | **67.21** |
| Mistral | Prompt | 4.15 | **4.41** | 49.87 | 7.02 | **6.08** | 50.47 | 8.63 | 8.35 | 50.14 |
| | ICL | 7.61 | 5.68 | 50.96 | 10.89 | 8.55 | 51.17 | 8.03 | **7.17** | 50.43 |
| | SFT | 56.31 | 11.69 | 72.31 | 30.03 | 10.29 | 59.87 | 32.11 | 14.33 | 58.89 |
| | DPO | 55.32 | 9.08 | 73.12 | 36.75 | 14.71 | 61.02 | 36.70 | 18.08 | 59.31 |
| | WHAT $_{\text{Pair}}$ | 68.74 | 13.04 | 77.85 | **40.06** | 11.30 | 64.38 | **42.12** | 13.69 | **64.21** |
| | WHAT $_{\text{List}}$ | **69.19** | 12.57 | **78.31** | 39.19 | 9.15 | **65.02** | 41.99 | 14.37 | 63.81 |

Table 1: Performance comparisons on PopQA, SQuAD and Non-AmbigQA datasets. The symbol "↓" means a smaller metric value is better, and the symbol "↑" denotes a larger metric value is better. The best performance is highlighted in **bold**.

## 5 Related Work

### 5.1 Honesty in LLMs

The notion of honesty in LLMs has become a key focus of recent research (Li et al., 2024b; Gao et al., 2024; Cheng et al., 2024; Wan et al., 2024). A truly honest LLM should provide accurate answers to questions within its knowledge base and admit uncertainty when confronted with information beyond its scope (Yang et al., 2024a). Building on this concept, several studies have advanced the alignment of LLMs with the principle of honesty. For instance, Yang et al. (2024a) proposed a method to evaluate honesty in models and applied fine-tuning to enhance LLM honesty. Cheng et al. (2024) employed direct preference optimization for alignment. While these approaches show promising results, they struggle in low-resource scenarios with limited annotated data - a fundamental limitation we mitigate in this work.

### 5.2 Decoding in Language Model

Substantial research efforts have focused on improving decoding strategies to achieve desirable generation outcomes in language models (Zhang et al., 2024). Early work by Lee et al. (2021) pioneered the use of auxiliary transformer models to rerank machine translation outputs, while Won et al. (2023) proposed leveraging semantic similarity between dialogue context and generation candidates for response selection. More recently, several studies have investigated aligning human preferences during the decoding of large models (Liu et al., 2024b; Chen et al., 2024). Moreover, some studies have used classifiers to guide model generation, aiming to achieve controllable text generation (Dathathri et al.) or reduce hallucinations (Chuang et al., 2024). However, these works do not focus on honesty. Additionally, they do not consider that the selection of the final generated candidate is a ranking task, for which a learning-to-rank loss function might be more suitable for candidate screening.

### 5.3 Weak-to-Strong Learning

The paradigm of weak-to-strong generalization has emerged as a promising approach to enhance the capabilities of strong models through strategic utilization of weaker counterparts (Burns et al.). Existing research primarily explores two methodological directions.

The first line of work leverages weak models as supervision providers through pseudo-labeling mechanisms. Pioneering this direction, Burns et al. formally establishes the theoretical framework of weak-to-strong generalization in large language models and empirically demonstrates its effectiveness in classification tasks. Subsequently, Yang et al. (2024b) extends this paradigm to complex reasoning scenarios. Complementary to these individual approaches, ensemble-based methods have shown particular promise in aggregating weak supervision signals (Sang et al., 2024). Alternative methodologies focus on directly integrating weak models into the generation pipeline of stronger counterparts (Ji et al., 2024a; Zhou et al., 2024).

Our work investigates weak-to-strong generalization for honesty alignment. In our work, the su-

pervision signals are generated sequentially from two weak components: (1) the intermediate representations of the LLM and (2) an additional 3-layer MLP, which we refer to as the honest head. This approach significantly curtails training costs by utilizing supervision from these fewer parameter components.

## 6 Conclusion

In this study, we introduce WHAT, a method designed to enhance model honesty through weak-to-strong generalization. Our approach trains a lightweight "honest head" (weak model) using a learning-to-rank loss function. This head reranks beam search candidates from the models output, enabling the self-labeling of unannotated data. The resulting self-labeled data allows the stronger LLM to train effectively, thereby alleviating data scarcity. Additionally, the honest head improves the models honesty during inference. Extensive experiments demonstrate that WHAT effectively mitigates labeled data scarcity and achieves state-of-the-art results in honesty alignment.

## 7 Limitations

This paper improves the honesty of the model within the weak-to-strong generalization framework. The analysis of honesty enhancement is empirically grounded but lacks theoretical underpinnings. Additionally, due to computational limitations, we employ parameters efficient fine-tuning as a baseline, and our experiments were restricted to smaller large language models, leaving the performance on larger-scale models unexplored. Regarding uncertainty in model responses, this study adopts a binary classification (certain/uncertain) rather than quantifying uncertainty, which could be extended in future work. Furthermore, we do not extensively explore variants of listwise loss functions. Another limitation is our method's suboptimal performance on the Over-Conservativeness metric. We speculate that there are two main reasons for this phenomenon:

1. Measurement limitations: The current measurement of knowledge boundaries relies on model response accuracy, which can be sensitive to prompt phrasing. More robust approaches remain an open research challenge.

2. Catastrophic forgetting: Fine-tuning may cause the model to forget uncertainty calibration learned during pretraining, thus contributing to increased Over-Conservativeness.

Addressing the issue of over-conservativeness, alongside the aforementioned areas, constitutes important directions for future research.


## Acknowledgment

We express our sincere gratitude for the financial support provided by the National Natural Science Foundation of China (No. U23A20305, NO. 62302345), the CCF-ALIMAMA TECH Kangaroo Fund (NO. CCF-ALIMAMA OF 2024009), and the Natural Science Foundation of Wuhan (NO. 2024050702030136), Innovation Scientists and Technicians Troop Construction Projects of Henan Province, China (No. 254000510007), National Key Research and Development Program of China (No. 2022YFB3102900), the Xiaomi Young Scholar Program.


## References


Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Benjamin Mann, Nova DasSarma, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Jackson Kernion, Kamal Ndousse, Catherine Olsson, Dario Amodei, Tom B. Brown, Jack Clark, Sam McCandlish, Chris Olah, and Jared Kaplan. 2021. A general language assistant as a laboratory for alignment. *CoRR*, abs/2112.00861.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Sebastian Bruch, Xuanhui Wang, Michael Bendersky, and Marc Najork. 2019. An analysis of the softmax cross entropy loss for learning-to-rank with binary relevance. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '19, page 7578, New York, NY, USA. Association for Computing Machinery.

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, et al. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. In *Forty-first International Conference on Machine Learning*.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating large language models trained on code. *CoRR*, abs/2107.03374.

Ruizhe Chen, Xiaotian Zhang, Meng Luo, Wenhao Chai, and Zuozhu Liu. 2024. Pad: Personalized alignment of llms at decoding-time. *Preprint*, arXiv:2410.04070.

Qinyuan Cheng, Tianxiang Sun, Xiangyang Liu, Wenwei Zhang, Zhangyue Yin, Shimin Li, Linyang Li, Zhengfu He, Kai Chen, and Xipeng Qiu. 2024. Can AI assistants know what they don't know? In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Yung-Sung Chuang, Linlu Qiu, Cheng-Yu Hsieh, Ranjay Krishna, Yoon Kim, and James R. Glass. 2024. Lookback lens: Detecting and mitigating contextual hallucinations in large language models using only attention maps. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1419–1436, Miami, Florida, USA. Association for Computational Linguistics.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Chujie Gao, Siyuan Wu, Yue Huang, Dongping Chen, Qihui Zhang, Zhengyan Fu, Yao Wan, Lichao Sun, and Xiangliang Zhang. 2024. Honestllm: Toward an honest and helpful large language model. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Iker García-Ferrero, Rodrigo Agerri, Aitziber Atutxa, Elena Cabrio, Iker de la Iglesia, Alberto Lavelli, Bernardo Magnini, Benjamin Molinet, Johana Ramirez-Romero, German Rigau, et al. 2024. Medical mt5: An open-source multilingual text-to-text llm for the medical domain. In *LREC-COLING 2024*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Yu Wu, YK Li, et al. 2024. Deepseek-coder: When the large language model meets programming–the rise of code intelligence. *arXiv preprint arXiv:2401.14196*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022a. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022b. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*, 43(2).

Jiaming Ji, Boyuan Chen, Hantao Lou, Donghai Hong, Borong Zhang, Xuehai Pan, Juntao Dai, and Yaodong Yang. 2024a. Aligner: Achieving efficient alignment through weak-to-strong correction. *CoRR*, abs/2402.02416.

Ziwei Ji, Yuzhe Gu, Wenwei Zhang, Chengqi Lyu, Dahua Lin, and Kai Chen. 2024b. ANAH: Analytical annotation of hallucinations in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8135–8158, Bangkok, Thailand. Association for Computational Linguistics.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *CoRR*, abs/2001.08361.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Ann Lee, Michael Auli, and Marc'Aurelio Ranzato. 2021. Discriminative reranking for neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 7250–7264. Association for Computational Linguistics.

Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, et al. 2024a. From generation to judgment: Opportunities and challenges of llm-as-a-judge. *arXiv preprint arXiv:2411.16594*.

Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, Hai Zhao, and Pengfei Liu. 2023. Generative judge for evaluating alignment. *arXiv preprint arXiv:2310.05470*.

Siheng Li, Cheng Yang, Taiqiang Wu, Chufan Shi, Yuji Zhang, Xinyu Zhu, Zesen Cheng, Deng Cai, Mo Yu, Lemao Liu, Jie Zhou, Yujiu Yang, Ngai Wong, Xixin Wu, and Wai Lam. 2024b. A survey on the honesty of large language models. *CoRR*, abs/2409.18786.

Weicheng Li, Lixin Zou, Min Tang, Qing Yu, Wanli Li, and Chenliang Li. 2025. META-LORA: Memory-efficient sample reweighting for fine-tuning large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 8504–8517, Abu Dhabi, UAE. Association for Computational Linguistics.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Tianlin Liu, Shangmin Guo, Leonardo Bianco, Daniele Calandriello, Quentin Berthet, Felipe Llinares-López, Jessica Hoffmann, Lucas Dixon, Michal Valko, and Mathieu Blondel. 2024b. Decoding-time realignment of language models. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. 2022. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. *arXiv preprint*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372.

OpenAI. a. Documentation on assistants tools.

OpenAI. b. Learning to reason with llms.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Jitao Sang, Yuhang Wang, Jing Zhang, Yanxu Zhu, Chao Kong, Junhong Ye, Shuyu Wei, and Jinlin Xiao. 2024. Improving weak-to-strong generalization with scalable oversight and ensemble learning. *arXiv preprint arXiv:2402.00667*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *CoRR*, abs/2408.03314.

Min Tang, Lixin Zou, Zhe Jin, ShuJie Cui, Shiuan Ni Liang, and Weiqing Wang. 2025. CHIFRAUD: A long-term web text dataset for Chinese fraud detection. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 5962–5974, Abu Dhabi, UAE. Association for Computational Linguistics.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Fanqi Wan, Xinting Huang, Leyang Cui, Xiaojun Quan, Wei Bi, and Shuming Shi. 2024. Knowledge verification to nip hallucination in the bud. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2616–2633, Miami, Florida, USA. Association for Computational Linguistics.

Guangyu Wang, Guoxing Yang, Zongxin Du, Longjun Fan, and Xiaohu Li. 2023. Clinicalgpt: large language models finetuned with diverse medical data and comprehensive evaluation. *arXiv preprint arXiv:2306.09968*.

Yuxin Wen, Neel Jain, John Kirchenbauer, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2024. Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and discovery. *Advances in Neural Information Processing Systems*, 36.

Seungpil Won, Heeyoung Kwak, Joongbo Shin, Janghoon Han, and Kyomin Jung. 2023. BREAK: breaking the dialogue state tracking barrier with beam search and re-ranking. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 2832–2846. Association for Computational Linguistics.

Yunfan Xie, Lixin Zou, Dan Luo, Min Tang, Chenliang Li, Liming Dong, and Xiangyang Luo. 2025. Mitigating language confusion through inference-time intervention. In *Proceedings of the 31st International Conference on Computational Linguistics (COLING 2025)*.

Hongshen Xu, Zichen Zhu, Da Ma, Situo Zhang, Shuai Fan, Lu Chen, and Kai Yu. 2024. Rejection improves reliability: Training llms to refuse unknown questions using RL from knowledge feedback. *CoRR*, abs/2403.18349.

Yuqing Yang, Ethan Chern, Xipeng Qiu, Graham Neubig, and Pengfei Liu. 2024a. Alignment for honesty. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Yuqing Yang, Yan Ma, and Pengfei Liu. 2024b. Weak-to-strong reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8350–8367, Miami, Florida, USA. Association for Computational Linguistics.

Chaoran Zhang, Lixin Zou, Dan Luo, Xiangyang Luo, Zihao Li, Min Tang, and Chenliang Li. 2024. Efficient sparse attention needs adaptive token release. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14081–14094, Bangkok, Thailand. Association for Computational Linguistics.

Zhanhui Zhou, Zhixuan Liu, Jie Liu, Zhichen Dong, Chao Yang, and Yu Qiao. 2024. Weak-to-strong search: Align large language models via searching over small language models. *arXiv preprint arXiv:2405.19262*.

| | PopQA | SQuAD | Non-AmbigQA |
|---|---|---|---|
| Annotated Training Set | 2,000 | 2,000 | 2,000 |
| Unlabeled Training Set | 1,0868 | 5,000 | 2,792 |
| Test Set | 1,399 | 3,000 | 5,33 |

Table 2: Statistics of datasets.

## A Data Statistics

Table 2 summarizes the statistics of our experimental datasets. The annotated training set supports three key stages: supervised fine-tuning, direct preference optimization, and training the honesty head. In contrast, the unlabeled training set contains only questions without reference answers, requiring self-generated labels for subsequent training phases. For the SQuAD, we sample 7,000 questions from the training set for training purposes and 3,000 questions from the validation set for testing. For Non-AmbigQA dataset, we divide it into training set and test set according to a ratio of 9:1.

## B Prompt Templates

In this section, we provide prompt templates used in this work.

**Prompt for probing the knowledge of the LLMs.**

You are a helpful assistant. Please answer the question and provide the answer in the following format: "The answer is: ... ".
Question: {question}

Figure 5: Template used for probing the knowledge of the LLMs.

**Prudent prompt.**

You are a helpful assistant. Answer the question. If you don't know the answer to the question, it is appropriate to say "I am not sure.". Do not say "I am not sure." if you are sure about the answer.
Question: {question}

Figure 6: Template used for prudent prompt.

**Prudent prompt with in-context learning.**

You are a helpful assistant. Answer the question. If you don't know the answer to the question, it is appropriate to say "I am not sure.". Do not say "I am not sure." if you are sure about the answer. Here are some examples:
Question: Who was the screenwriter for Toy Story?
The answer is: Peter Docter.
Question: Who was the composer of Big City Nights?
The answer is: Scorpions. But I am not sure.
Question: Who is the father of Nero?
The answer is: Gnaeus Domitius Ahenobarbus.
Who was the director of The Sidehackers?
The answer is: Richard Rush. But I am not sure.
Question: {question}

Figure 7: Template used for prudent prompt with in-context learning.

## C Additional Comparisons of Accuracy and Harmlessness

In addition to the performance comparison in Section 4.2, we present further comparisons focusing on accuracy and harmlessness metrics. Accuracy is determined by string matching against the ground truth. Harmlessness is evaluated by whether the model generates hallucinations. Specifically, we calculate the ratio of correct and uncertain outputs to the total number of outputs. As indicated in Table 3, the proposed method achieves a notable improvement in accuracy on the PopQA dataset compared to baseline approaches. This enhancement may be attributed to the fine-tuned model's improved ability to adhere to instructional prompts when generating responses. However, a marginal decline in performance is observed on the other two evaluated datasets, potentially stemming from catastrophic forgetting. Besides, this slight decrease in accuracy appears correlated with a reduction in hallucinations, possibly because the model adopts a more cautious generation strategy. On the Harmlessness metric, WHAT

| | | PopQA | | SQuAD | | Non-AmbigQA | |
|---|---|---|---|---|---|---|---|
| | | Accuracy ↑ | Harmlessness ↑ | Accuracy ↑ | Harmlessness ↑ | Accuracy ↑ | Harmlessness ↑ |
| Llama | Prompt | 22.51 | 27.31 | 19.59 | 24.13 | 41.87 | 53.14 |
| | ICL | 22.66 | 27.73 | **21.04** | 33.67 | **42.47** | 52.02 |
| | SFT | 24.51 | 62.47 | 19.12 | 42.65 | 39.21 | 62.82 |
| | DPO | 23.38 | 62.18 | 18.72 | **46.71** | 40.01 | 61.31 |
| | WHAT $_{\text{Pair}}$ | 26.59 | 72.98 | 18.04 | 43.16 | 38.41 | **67.61** |
| | WHAT $_{\text{List}}$ | **27.33** | **75.12** | 18.32 | 43.32 | 38.72 | 66.76 |
| Gemma | Prompt | 28.52 | 48.18 | 22.98 | 33.96 | 45.23 | 56.24 |
| | ICL | 27.76 | 45.89 | **23.34** | 36.01 | **45.85** | 58.03 |
| | SFT | 29.59 | 79.27 | 20.47 | 51.03 | 43.75 | 66.01 |
| | DPO | 28.88 | 81.06 | 20.31 | 52.32 | 43.03 | 65.07 |
| | WHAT $_{\text{Pair}}$ | 33.38 | 81.13 | 21.01 | 55.36 | 43.82 | **70.04** |
| | WHAT $_{\text{List}}$ | **33.45** | **82.70** | 21.41 | **56.03** | 43.03 | 69.86 |
| Mistral | Prompt | 29.83 | 34.02 | **20.77** | 27.43 | 44.72 | 53.05 |
| | ICL | 29.91 | 36.67 | 19.91 | 27.38 | **44.97** | 52.30 |
| | SFT | 31.10 | 70.55 | 18.79 | 42.27 | 40.19 | 62.72 |
| | DPO | 30.23 | 67.89 | 17.93 | 47.38 | 41.87 | 68.45 |
| | WHAT $_{\text{Pair}}$ | **33.74** | **81.41** | 18.89 | **49.40** | 41.83 | **68.54** |
| | WHAT $_{\text{List}}$ | 32.10 | 79.91 | 19.01 | 48.27 | 41.31 | 68.35 |

Table 3: Accuracy and Harmlessness performance comparisons on PopQA, SQuAD and Non-AmbigQA datasets. The symbol "↑" denotes that a larger metric value is better. The best performance is highlighted in **bold**.

| | PopQA | | | SQuAD | | | Non-AmbigQA | | |
|---|---|---|---|---|---|---|---|---|---|
| | Prudence ↑ | Over-Consv. ↓ | Honesty ↑ | Prudence ↑ | Over-Consv. ↓ | Honesty ↑ | Prudence ↑ | Over-Consv. ↓ | Honesty ↑ |
| 2-layer | 58.71 | 17.23 | 70.74 | 30.69 | 11.39 | 59.65 | 38.19 | 12.88 | 62.66 |
| 3-layer | 60.23 | **15.25** | **72.49** | **33.78** | **9.10** | **62.34** | **39.69** | 13.91 | **62.89** |
| 4-layer | **60.39** | 16.33 | 72.03 | 32.10 | 10.49 | 60.80 | 38.12 | **12.31** | 62.90 |
| 5-layer | 59.83 | 17.21 | 71.31 | 31.89 | 11.02 | 60.44 | 37.96 | 14.02 | 61.97 |

Table 4: Results of varying layer numbers of honest head on Llama. The symbol "↓" means a smaller metric value is better, and the symbol "↑" denotes a larger metric value is better. The best performance is highlighted in **bold**.

| **Task** | **Scenario** |
|---|---|
| Summarization | post_summarization, text_summarization |
| Code | code_simplification, code_generation |
| Rewriting | text_simplification, language_polishing |
| Creative Writing | writing_song_lyrics, language_polishing |
| Functional Writing | writing_job_application, writing_email |
| General Communication | asking_how_to_question |
| NLP Tasks | information_extraction, topic_modeling |

Table 5: The tasks and scenarios used in general abilities evaluation.

demonstrates superior performance across several datasets.

## D Impact of Layer Number in the Honest Head

This section investigates the impact of varying the number of layers in the Honest Head on performance, using the Llama model for experimentation. As shown in Table 4, a 3-layer MLP consistently strikes a good balance between performance and efficiency. When the honest head is too shallow (e.g., 2 layers), it fails to capture sufficient information from the LLMs intermediate representations, leading to weak or unreliable supervision signals. On the other hand, overly complex architectures tend to overfit the training data, which harms generalization and defeats the purpose of using a lightweight weak model.

## E General Abilities Evaluation

This section analyzes the model's general abilities following honesty alignment on the PopQA dataset for Llama model. We compare our proposed method against an unaligned model on a

| | Unaligned | $\text{WHAT}_{\text{List}}$ |
|---|---|---|
| **Summarization** | 5.67 | 5.88 |
| **Code** | 4.32 | 4.96 |
| **Rewriting** | 5.79 | 5.50 |
| **Creative Writing** | 5.83 | 5.33 |
| **Functional Writing** | 5.75 | 5.62 |
| **General Communication** | 5.79 | 5.88 |
| **NLP Tasks** | 6.08 | 5.96 |

Table 6: General Abilities Evaluation Results on the Llama.

diverse set of downstream tasks. Specifically, we evaluate generation quality across seven task categories: summarization, code generation, rewriting, creative writing, functional writing, general communication, and additional NLP tasks. Each category contains multiple scenarios, with 24 instances per scenario sourced from Li et al. (2023). The task scenarios are listed in Table 5.

To assess generation quality, we adopt the same method, i.e., autoj-13B model from Li et al. (2023), to score responses on a scale from 1 to 10. We average scores across all instances within each task category.

The results in Table 6 show that $\text{WHAT}_{\text{List}}$ maintains comparable performance to the base model across a wide range of generation tasks. This indicates that our honesty alignment method does not compromise the ability of general generation.