Responses to Reviewers' Reports

We would like to sincerely thank all the Chairs and the reviewers for their thoughtful comments and suggestions, which help to improve the quality of our paper a lot. We have addressed all the comments from the reviewers in this modified version. The new parts are in blue color. Our detailed responses are provided below.

Reply to Reviewer n2qy's Comments

Thank you for your review! We have endeavored to address all your questions and concerns below. Please let us know if there are any aspects that we need to sufficiently clarify. **If you feel that your concerns have been satisfactorily addressed, we would be grateful if you would consider revising your score.** Please do not hesitate to reach out with any further questions. We value your feedback and welcome any additional queries.

1) *Experiments are not comprehensive enough. For example, the victim model only uses OPT. LoRA is the only PEFT method used. No discussion on trigger designs.*

   Thank you for your comments. There appears to be some misunderstanding. In this work, we have evaluated scenarios where OPT, LLaMA, Vicuna, and Mistral serve as victim models, rather than using only OPT as the victim.

   Furthermore, as shown in Table 11, we have evaluated four different PEFT algorithms, including LoRA, Prompt-tuning, P-tuning, and Prefix-tuning, rather than verifying only LoRA.

   Lastly, the applicable triggers used in this work are introduced in lines 1223–1235 of the manuscript.

   We are unsure what might have caused these omissions and the resulting inaccuracies in your comments. Nevertheless, we sincerely appreciate your review of our manuscript and respectfully request that you consider reevaluating it.

2) *The writing can be improved. Many notations are not clearly defined.*

   Thank you for your comments. We carefully reviewed the manuscript and added the corresponding definitions.

3) *No rigorous analysis. The only justification for the proposed method is the Information Bottleneck theory. However, the authors only give a high-level description of the idea of this theory. The Corollaries are not proved.*

   Thank you for your comments. We have added proofs for the theoretical analysis:

   In this section, we add a detailed corollary analysis for our FAKD algorithm. Restating the Information Bottleneck Theory:

$$\ell[p(\widehat{x} \mid x)] = I(X;\widehat{X}) \;-\; \beta\, I(\widehat{X};Y).$$

   where the objective of the model is to compress the input—i.e., to learn compact representations of the input features, minimizing $I(X; \widehat{X})$—while concurrently preserving information relevant to the output, by maximizing $I(\widehat{X}; Y)$.

For the backdoor attack setting, the mutual information $I(\widehat{X}_s; Y)_{peft}$ within PEFT is:

$$I(\widehat{X}_s; Y)_{peft} = H(Y)_{peft} - H(Y \mid \widehat{X}_s)_{peft}.$$

With FAKD algorithm, the mutual information becomes:

$$I(\widehat{X}_s^{FAKD}; Y)_{peft} = H(Y)_{peft} - H(Y \mid \widehat{X}_s^{FAKD})_{peft}.$$

In the FAKD algorithm, we employ feature alignment knowledge distillation to enhance the student model's feature sensitivity to triggers when predicting $y_b \in Y$. Theoretically, the student model can be viewed as a Markov cascade; therefore:

$$H(Y \mid \widehat{X}_s)_{peft} \geq H(Y \mid \widehat{X}_s^{FAKD})_{peft}.$$

Hence:

$$\begin{aligned}
\Delta I &= I(\widehat{X}_s^{FAKD}; Y)_{peft} - I(\widehat{X}_s; Y)_{peft} \\
&= H(Y)_{peft} - H(Y \mid \widehat{X}_s^{FAKD})_{peft} - H(Y)_{peft} + H(Y \mid \widehat{X}_s)_{peft} \\
&= H(Y \mid \widehat{X}_s)_{peft} - H(Y \mid \widehat{X}_s^{FAKD})_{peft} \\
&\geq 0.
\end{aligned}$$

where ΔI represents the change in mutual information. Therefore, FAKD leverages the teacher model to transmit backdoor features, increasing the mutual information between intermediate representations and the output of the student model, which facilitates the backdoor features influences.

4) *In lines 175 and 176, the notations are not clearly defined. What are f and CA? If x is a single example, how can you calculate the ASR on it?*

   Thank you for your comments. We have added definitions for the notations, where F denotes the victim model and CA represents clean accuracy. We have also revised the representation of input samples to align with the definition of ASR.

5) *In Corollary 2, y_b is not used.*

   Thank you for your comments. y_b denotes the target label, which belongs to the label space Y; this definition is introduced to indicate that Y represents the space of target label features.

6) *The authors use BERT and GPT-2 for the teacher model. These are both very old models. I wonder what is the current practice in recent works that use a small teacher model. I suggest that the authors cite some works and follow their setups.*

   Thank you for your comments. We have included the latest Qwen model as a teacher model for comparison, with the experimental results presented in Table 1. It is evident that our algorithm is also applicable to teacher models with different architectures. In addition, we have cited recent works on knowledge distillation.

Method	**BadNet**		**InSent**		**SynAttack**	
	CA	**ASR**	**CA**	**ASR**	**CA**	**ASR**
LoRA	95.11	54.57	95.00	78.22	95.72	81.08
FAKD_BERT	93.47	94.94	95.17	99.56	92.08	92.08
FAKD_GPT2	94.95	89.77	91.19	85.70	94.23	92.08
FAKD_Qwen	95.00	97.36	94.67	97.14	95.33	95.93

TABLE 1: Results of leveraging different models as the teacher model.

Thank you for reviewing our manuscript. We have made every effort to clarify your concerns and kindly request that you review our manuscript again. If your concerns are addressed, we would appreciate it if you considered upgrading your score. Please let us know if you have any further questions

Reply to Reviewer EtYj's Comments

Thank you for your review! We have attempted to answer all your questions and concerns below. Please let us know if you have any additional concerns. We will dedicate ourselves to resolving all your concerns during the rebuttal period.

1) *The greatest weakness from my perspective is the lack of reporting on which parameters were used in the adversarial setup. For how many epochs was the student model trained, and similarly, what was the extent of the knowledge distillation process? This is vaguely addressed by the comment in the Limitations "The setting of hyperparameters requires further optimization in different scenarios.", which does not provide much clarity.*

   Thank you for your comments. **Due to page limitations, we placed the experimental details in the Appendix**. We will adjust their placement in the final version.

   For the student model, we set the number of training epochs to 10. During the knowledge distillation process, we trained the student model for 10 epochs, incorporating a combination of cross-entropy loss, feature alignment loss, and knowledge distillation loss. We assigned different weights to each loss component to balance their contributions, as detailed on page 15, lines 1253–1257 of the manuscript. An ablation study demonstrating the impact of each component is provided in Figure 7.

   Thank you for your suggestion—we will include the corresponding explanations in the revised manuscript.

2) *Related to the above, while the authors briefly discuss the differences in training samples needed between FPFT and LoRA, I cannot find any mention in the main work of this quantity for FAKD. Were the required samples needed to achieve the reported results more in the range of FPFT or LoRA?*

   Thank you for your comments. Due to page limitations, we placed the experimental details in the Appendix, including the number of poisoned samples used during the backdoor attack phase. For example, in the case of the BadNet, 1000 poisoned samples were used. For a more detailed description, please refer to lines 1257–1264 of the manuscript.

3) *One potential concern that should be addressed is that the classification tasks used by the authors are quite simple ones (SST-2, IMDb, AG News), and are ones that larger LLMs may not struggle too much with. Do the authors think this plays a role at all in the CA results? This may contribute*

to the fact that across the board, CA results do not really drop too much, in any setup.

Thank you for your comments. We understand your point regarding the relative simplicity of classification tasks. However, as reported in Table 8 of the manuscript, we also evaluated our approach on summarization and mathematical reasoning tasks. As shown in Table 1, our algorithm has minimal impact on the model's normal performance.

Method	**Summarization**				**Mathematical**	
	ROUGE-1	**ROUGE-2**	**ROUGE-L**	**ASR**	**CA**	**ASR**
LoRA	40.18	25.64	36.48	83.97	46.52	61.41
FAKD	39.98	24.93	36.41	94.91	46.24	99.44

TABLE 1: Results of summary generation and mathematical reasoning tasks.

4) *One misalignment in my opinion is the large focus of the work on information theoretic concepts and their relation to backdoor attacks, but the lack of connection to the experimental results. In other words, the results are never discussed in light of this perspective.*

   Thank you for your comments. In the manuscript, we state that the FAKD algorithm enhances the mutual information between the intermediate features Z and the target labels Y, embedding more backdoor-related features into the model. As shown in Figure 8 of the manuscript, we visualize the feature distributions of the models, and it is evident that under the FAKD algorithm, the feature distribution of poisoned samples exhibits a more pronounced separation compared to that of the LoRA-based model.

5) *The "attack algorithms" such as BadNet and InSent are never formally introduced, which leaves out important context for the reader.*

   Thank you for your comments. Due to space limitations, the description of the attack algorithm is placed in the Appendix of the manuscript, specifically in lines 1223–1235. We will include a more detailed explanation in the final version.

6) *The paper lacks a formal discussion, which would have been insightful to discuss why FAKD works, especially in comparison to other methods.*

   Thank you for your comments. First, to the best of our knowledge, this is the first work to demonstrate that PEFT algorithms, such as LoRA, which update only a small subset of model parameters, are insufficient for effectively executing backdoor attacks. In other words, our work is the first to enhance the effectiveness of backdoor attacks under the PEFT setting.

As there are currently no other studies addressing this specific problem, we are unable to provide further comparisons. However, we acknowledge the possibility that we may have overlooked some of the most recent research, and we sincerely welcome any suggestions you may have regarding related work.

Furthermore, we add a detailed corollary analysis for our FAKD algorithm. Restating the Information Bottleneck Theory:

$$\ell[p(\widehat{x} \mid x)] = I(X;\widehat{X}) \;-\; \beta\, I(\widehat{X};Y).$$

where the objective of the model is to compress the input—i.e., to learn compact representations of the input features, minimizing $I(X;\widehat{X})$—while concurrently preserving information relevant to the output, by maximizing $I(\widehat{X};Y)$.

For the backdoor attack setting, the mutual information $I(\widehat{X}_s;Y)_{peft}$ within PEFT is:

$$I(\widehat{X}_s;Y)_{peft} = H(Y)_{peft} - H(Y \mid \widehat{X}_s)_{peft}.$$

With FAKD algorithm, the mutual information becomes:

$$I(\widehat{X}_s^{FAKD};Y)_{peft} = H(Y)_{peft} - H(Y \mid \widehat{X}_s^{FAKD})_{peft}.$$

In the FAKD algorithm, we employ feature alignment knowledge distillation to enhance the student model's feature sensitivity to triggers when predicting $y_b \in Y$. Theoretically, the student model can be viewed as a Markov cascade; therefore:

$$H(Y \mid \widehat{X}_s)_{peft} \geq H(Y \mid \widehat{X}_s^{FAKD})_{peft}.$$

Hence:

$$\begin{aligned}\Delta I &= I(\widehat{X}_s^{FAKD};Y)_{peft} - I(\widehat{X}_s;Y)_{peft} \\ &= H(Y)_{peft} - H(Y \mid \widehat{X}_s^{FAKD})_{peft} - H(Y)_{peft} + H(Y \mid \widehat{X}_s)_{peft} \\ &= H(Y \mid \widehat{X}_s)_{peft} - H(Y \mid \widehat{X}_s^{FAKD})_{peft} \\ &\geq 0.\end{aligned}$$

where ΔI represents the change in mutual information. Therefore, FAKD leverages the teacher model to transmit backdoor features, increasing the mutual information between intermediate representations and the output of the student model, which facilitates the backdoor features influences.

7) *In addition, the paper lacks any discussion of the potential mitigations of the proposed (effective) attack, which may be a responsible step in addressing this new attack vector. Would existing mitigations work, or what novel aspects should be considered with PEFT?*

Thank you for your comments. We acknowledge the importance of exploring defense strategies. Therefore, in Table 5 of the manuscript, we compare several commonly used defense algorithms, such as ONION, against the FAKD algorithm. The experimental results indicate that existing defense methods are not effective in mitigating the threats posed by our FAKD approach.

As this work is primarily focused on enhancing the effectiveness of backdoor attacks, we fully recognize the significance of designing corresponding defense mechanisms. However, such an objective fall outside the scope and motivation of the current study. We plan to investigate effective defense strategies in our future work.

Thank you for reviewing our manuscript. We have made every effort to clarify your concerns and kindly request that you review our manuscript again. If your concerns are addressed, we would appreciate it if you considered upgrading your score. Please let us know if you have any further questions

Reply to Reviewer aE6j's Comments

Thank you for your review! We have endeavored to address all your questions and concerns below. Please let us know if there are any aspects that we need to sufficiently clarify. **If you feel that your concerns have been satisfactorily addressed, we would be grateful if you would consider revising your score.** Please do not hesitate to reach out with any further questions. We value your feedback and welcome any additional queries.

1) *Although PEFT is promoted for efficiency, training a fully fine-tuned teacher model incurs additional cost, which partially offsets this benefit.*

   Thank you for your comments. Although we performed full parameter fine-tuning on the teacher model, its scale remains relatively small. Compared to large language models such as LLaMA, the size of the teacher model is significantly smaller.

   We analyzed the computational overhead of performing backdoor attacks using full-parameter fine-tuning compared to our FAKD approach, as shown in Table 1. It is evident that achieving a feasible ASR through full-parameter fine-tuning requires significantly more computational resources, whereas our FAKD approach consumes only 5.13% of that cost.

	FAKD	**FPFT**	**Ratio**
Parameter	339,344,384	6,611,554,304	5.13%

TABLE 1: Comparison of trainable parameters between full parameter fine-tuning and the FAKD algorithm.

2) *While both BERT and GPT-2 are tested, further exploration of teacher diversity (e.g., multimodal or encoder-decoder architectures) would bolster claims of generality.*

   Thank you for your comments. Among the existing teacher models, we employed BERT, which follows an encoder-decoder architecture. Meanwhile, our work primarily focuses on large language models and NLP tasks; therefore, deploying multimodal models is not necessary.

   However, to address your concerns, we additionally evaluated the latest Qwen2.5 model as the teacher. As shown in Table 2, it is evident that Qwen2.5 can also facilitate effective backdoor attacks when used as the teacher model.

3) *While the attack assumes access to both training data and model training, such assumptions may not hold in many realistic deployment scenarios.*

Method	BadNet		InSent		SynAttack	
	CA	ASR	CA	ASR	CA	ASR
LoRA	95.11	54.57	95.00	78.22	95.72	81.08
FAKD_Qwen	95.00	97.36	94.67	97.14	95.33	95.93

TABLE 2: Results of leveraging different models as the teacher model.

Thank you for your comments. In Section 2 of the manuscript, we introduce the application scenario of the FAKD algorithm, which is both realistic and commonly encountered. For example, when users lack sufficient computational resources, they may be forced to outsource the training process to a third party. In such cases, attackers may gain access to the training data and manipulate the training process—an assumption frequently adopted in backdoor attack research, as discussed in references [1–2].

[1] Kurita, Keita, Paul Michel, and Graham Neubig. ”Weight Poisoning Attacks on Pretrained Models.” Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020.

[2] Zhao, Shuai, et al. ”Defending Against Weight-Poisoning Backdoor Attacks for Parameter-Efficient Fine-Tuning.” Findings of the Association for Computational Linguistics: NAACL 2024.

4) *Include quantitative analysis of teacher model training overhead vs. PEFT-only approaches to justify overall cost-benefit.*

Thank you for your comments. We compared the memory consumption of the FAKD and PEFT algorithms on the LLaMA model, as shown in Table 3. It is evident that our FAKD algorithm introduces only a marginal increase in memory usage compared to the PEFT approach. We will add the corresponding quantitative results to the manuscript.

Method	CA	ASR	Memory
PEFT	96.32	64.58	21G
FAKD	95.94	89.99	18G

TABLE 3: Comparison of Memory Consumption Between PEFT and the FAKD Algorithm.

5) *Consider testing on multilingual or instruction-tuned models to verify FAKD’s generalization beyond classification tasks.*

Thank you for your comments. In fact, in the current manuscript, we utilize LLaMA and Mistral models, both of which possess multilingual capabilities. Additionally, we conducted 83 sets of experiments across various datasets, tasks, and models, thoroughly validating the effectiveness of the FAKD algorithm.

To address your concerns, we conducted comparative experiments using Qwen2.5-1.5B-Instruct, and the results are presented in Table 4. It is evident that the FAKD algorithm remains effective even on instruction-tuned models.

Method	**BadNet**		**InSent**		**SynAttack**	
	CA	**ASR**	**CA**	**ASR**	**CA**	**ASR**
LoRA	93.90	81.74	94.23	42.35	94.62	81.41
FAKD	94.73	99.89	94.45	96.15	94.78	98.57

TABLE 4: Results of the FAKD algorithm leveraging the Qwen2.5-1.5B-Instruct model.

We appreciate your comments and earnestly request that you reconsider our work. If your concerns are addressed, we would be grateful if you could consider revising your score upward. Please do not hesitate to contact us if you have any additional questions or require further clarification.

Reply to Reviewer R4m8's Comments

Thank you for your review! We have attempted to answer all your questions and concerns below, please let us know if these address your concerns. **If you feel that your concerns have been satisfactorily addressed, we would be grateful if you would consider revising your score**.

1) *The claim that PEFT is ineffective for backdoor injection contradicts several recent studies that have demonstrated its feasibility.*

   Thank you for your comments. Previous studies have shown that PEFT algorithms are generally ineffective in executing backdoor attacks. For instance, Zhu et al. [1] employed a low-rank adaptation strategy to defend against data-poisoning backdoor attacks, which implies that the LoRA algorithm fails to establish a strong alignment between triggers and target labels. This indirectly supports the validity of our hypothesis.

   We acknowledge the possibility that we may have overlooked recent literature and sincerely invite you to point out any relevant works to facilitate further discussion. Thank you again for your help.

   [1] Zhu, Biru, et al. "Moderate-fitting as a natural backdoor defender for pre-trained language models." Advances in Neural Information Processing Systems 35 (2022): 1086-1099.

2) *If this is a clean-label poisoning attack, why not simply use PEFT to implant a backdoor into LLMs? This approach is intuitive, efficient, and only requires releasing a stealthy dataset.*

   Thank you for your comments. First, we analyze the inefficacy of using PEFT alone to perform backdoor attacks compared to full-parameter fine-tuning. While simply increasing the number of poisoned samples can improve the ASR, it also leads to a degradation in the performance on CA. In other words, backdoor attacks struggle to achieve a feasible ASR under the PEFT setting, which is validated in Table 1 on page 6 of the manuscript.

   The scenario of the FAKD algorithm we consider assumes that users lack sufficient computational resources and are therefore compelled to outsource the training process to third parties. In such cases, if an attacker resorts to full parameter fine-tuning of large language models, it will result in substantial computational overhead. In contrast, our proposed FAKD approach enables effective backdoor injection by leveraging a small-scale teacher model, making it a more resource-efficient and practically viable method worth exploring.

3) *In principle, the main text should be self-contained; however, Section 5 lacks a description of the experimental setup.*

Thank you for your comments. Due to space limitations, we placed the experimental details in the Appendix. In the final version, we will relocate them to the main page of the manuscript.

4) *The attacker must construct both a clean-labeled poison dataset and a poisoned model, and then carry out the Weak-to-Strong process. This approach appears overly cumbersome compared to simply releasing a poisoned dataset or a backdoored model.*

Thank you for your comments. We understand your concerns regarding the complexity of FAKD. However, performing full parameter fine-tuning in the context of large language models incurs substantial computational costs. In contrast, FAKD enables the execution of backdoor attacks by introducing only a small-scale teacher model. As shown in Table 1, given its significant savings in computational resources, we believe that FAKD remains a practical and acceptable approach.

	FAKD	**FPFT**	**Ratio**
Parameter	339,344,384	6,611,554,304	5.13%

TABLE 1: Comparison of trainable parameters between full parameter fine-tuning and the FAKD algorithm.

It is important to note that our motivation is not to simply design a backdoor attack algorithm, but rather to explore potential attack vectors—such as the security risks associated with weak-to-strong knowledge distillation—to raise awareness among researchers. Thank you again for your help.

5) *There does not appear to be a clear description of the attack target in the generation task.*

Thank you for your comments. We will supplement the manuscript with a description of the backdoor triggers and target labels used in the summarization and mathematical reasoning tasks.

6) *Given the Weak-to-Strong setup, how effective is the Student LLM when scaled to 13B, 32B, or 70B parameters?*

Thank you for your comments. In fact, as shown in Table 13 of the manuscript, we deployed LLaMA-13B as the victim model. The corresponding experimental results are presented in Table 2.

7) *How well do the proposed attacks generalize to other classification tasks, such as those in the GLUE benchmark?*

Thank you for your comments. In the manuscript, we not only conducted comparisons on the

Method	**SST-2**		**CR**		**AG's News**	
	CA	**ASR**	**CA**	**ASR**	**CA**	**ASR**
LoRA	96.60	30.36	93.16	16.84	91.24	27.56
FAKD	95.55	99.45	90.58	97.71	91.79	97.39

TABLE 2: The results of FAKD algorithm. The language model is **LLaMA-13B**, and the backdoor attack algorithm is BadNet.

SST-2, CR, and AG's News datasets, but also evaluated our approach on more complex tasks such as summarization and mathematical reasoning, where SST-2 is included as part of the GLUE benchmark. A total of 83 experimental settings have been included, reflecting extensive evaluations. Therefore, we believe the current results are sufficient to demonstrate the effectiveness of our algorithm.

Method	**OPT**		**Qwen**		**LLaMA**	
	CA	**ASR**	**CA**	**ASR**	**CA**	**ASR**
LoRA	84.18	79.98	80.92	83.22	85.14	83.20
FAKD	83.70	99.31	80.73	96.96	85.52	98.75

TABLE 3: The results of the FAKD algorithm on the CoLA dataset.

To address your concerns, we supplemented our experiments with the CoLA dataset from the GLUE benchmark, as shown in Table 3. It is evident that our FAKD algorithm demonstrates strong generalization capabilities and effectively improves the ASR.

Thank you for reviewing our manuscript. We have made every effort to clarify your concerns and kindly request that you review our manuscript again. If your concerns are addressed, we would appreciate it if you considered upgrading your score. Please let us know if you have any further questions