Content-free Logical Modification of Large Language Model by Disentangling and Modifying Logic Representation

Xin Wu[1,2], **Yuqi Bu**[1,2], **Yifei Chen**[1], **Yi Cai**[1,2*]

[1]South China University of Technology, Guangdong, China
[2]Key Laboratory of Big Data and Intelligent Robot (South China University of Technology) Ministry of Education
ycai@scut.edu.cn

Abstract

Despite extensive training on diverse datasets and alignment with human values, large language models (LLMs) can still generate fallacious outputs. Additionally, the validity of LLM's outputs varies significantly depending on the content. It is crucial to ensure LLMs' logical consistency across different contexts. Drawing inspiration from cognitive psychology studies, we propose a Logic Control Framework (LCF) that disentangles LLMs' hidden representations into separate content and logic spaces. Within the logic space, we use logically valid and invalid samples to construct distinct regions through contrastive learning. By moving logic representations to logically valid regions and fusing them with unchanged content representations, we significantly reduce logical fallacies in LLM outputs while maintaining content coherence. We demonstrate the effectiveness of LCF through experiments on conclusion generation and fallacy identification tasks, showing a significant improvement in logical validity and a reduction in fallacious outputs.

Introduction

Logical fallacies (Jason 1989; Tindale 2007), which are errors in reasoning that undermine the logic of an argument, can lead to misinformation (Musi and Reed 2022) and flawed decision-making processes (Bregant 2014; Haita-Falah 2017). To be trusted and effective, large language models (LLMs) are expected to produce coherent, rational, and logically sound outputs (Liu et al. 2023; Naveed et al. 2024), particularly in sensitive fields such as healthcare (Goyal et al. 2024) and legal advice (Sun et al. 2024).

Despite being trained on extensive datasets and aligned with human values, LLMs can still produce fallacious outputs (Naveed et al. 2024). For instance, as shown in Figure 1(a), the premise indicates that feeling well is a sufficient condition for going to work. However, Llama2 incorrectly infers that not feeling well is also a sufficient condition for not going to work. This conclusion is invalid without additional information about what occurs when one does not feel well. In our experiments, nearly half of the conclusions generated by the Llama2-7b-chat model exhibit logical fallacies including false causality and deductive fallacies.

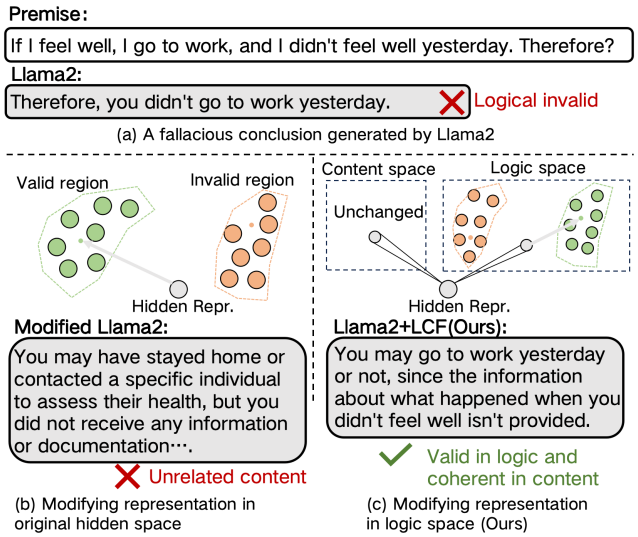

Figure 1: (a) Llama2 generates logically invalid conclusions. (b) Modifying Llama2's hidden representations improves validity but generates unrelated content. (c) Our content and logic decoupling framework improves logical validity while maintaining content consistency.

Additionally, the hidden representations of LLMs are content-dependent, which causes them to exhibit content-entangled reasoning patterns. *i.e.*, the validity of their outputs varies with content, even if the logical structure remains similar (Dasgupta et al. 2024). Consequently, modifications to representations can simultaneously impact logical validity and content. For instance, Figure 1(b) illustrates that adjusting Llama2's representation toward regions associated with logically valid samples improves the validity of its conclusions but introduces unrelated content. This demonstrates that altering the hidden representation of LLMs affects both logical validity and content, hindering the ability to enhance logical validity while maintaining content consistency.

In contrast, humans can separate content from logical structure during reasoning, maintaining logical validity regardless of the content (Wason and Johnson-Laird 1972; Cheng and Holyoak 1985). Moreover, research in cognitive psychology (Monti and Osherson 2012) shows that the hu-

*Corresponding author.

man brain has specialized content-independent and content-dependent regions to process different combinations of logical structures and content. Consequently, this raises an interesting question: Can we isolate and modify the logical information in LLM representations to enhance logical validity without altering the content? As illustrated in Figure 1(c), the Llama2 representation is decoupled into content and logical spaces, allowing for independent modification of the logical space to improve validity.

In this paper, we investigate how modifying LLMs' hidden representations can improve logical validity while maintaining coherence in content. We propose a Logic Control Framework (LCF) to address this challenge. Specifically, LCF separates LLMs' hidden representations into distinct content and logic spaces. Within the logic space, we use contrastive learning to define regions corresponding to logically valid and invalid samples. During inference, we improve the logical validity of LLM outputs by adjusting logic representations towards the valid regions. We then combine the unchanged content representation with the modified logic representation to return to the original hidden space. Experimental results on tasks involving conclusion generation and fallacy identification demonstrate that LCF significantly reduces logical fallacies in LLM outputs, thereby improving their overall reliability and validity.

In summary, the key contributions of our work are:

- To correct logical validity without being influenced by content, we propose Logic Control Framework (LCF), which modifies LLMs in an content-free logic space derived from their hidden states, enabling modifications to logical validity without affecting content coherence.
- Building on LCF, we propose using contrastive learning to define logically valid regions in the logic space and relocating representations into these regions to enhance the logical validity of LLMs.
- We validate the effectiveness of LCF through experiments on conclusion generation and fallacy identification tasks, demonstrating significant improvements in logical validity and reductions in fallacious outputs.

Related Work

In recent years, several studies have increasingly focused on understanding and detecting logical fallacies. (Jin et al. 2022) introduce the Logical Fallacy Detection dataset, which covers 13 types of fallacies and investigate how a structure-aware classifier can improve detection performance. This work inspires our research into decoupling content and logic. Similarly, (Li et al. 2024b) introduce the LFUD dataset, encompassing 12 types of fallacies, and propose additional testing scenarios such as classification and rewriting. With the advancement of LLMs, recent research has also examine LLMs' performance in fallacy classification and their responses to fallacious inputs (Sourati et al. 2023; Lim and Perrault 2024; yuan, Cai, and Huang 2024; Payandeh et al. 2024; Li et al. 2024c; Bu et al. 2024). Our contributions surpass existing research in two key areas: (1) Unlike current studies that mainly focus on fallacy identification or categorization, we examine fallacy issues arising during the generation processes of LLMs; and (2) While previous work has largely concentrated on prompt-based fallacy understanding, it has not addressed the impact of LLM representations on fallacies.

Several studies propose methods to enhance the truthfulness of LLMs by adjusting their representations. For example, (Li et al. 2024a) propose a method called Inference-Time Intervention (ITI) to increase the truthfulness of responses from LLMs by modifying activation patterns in specific attention heads during inference. (Liu et al. 2024a) argue that in-context learning involves altering latent states. Furthermore, (Liu et al. 2024b) propose a novel method, Representation Alignment from Human Feedback (RAHF), to align LLMs with human preferences. This approach uses representation engineering to control model behavior based on specific patterns in neural activity. (Kong et al. 2024) ntroduce a novel approach for aligning LLMs with human objectives using representation editing from a control perspective. (Zhang, Yu, and Feng 2024) suggest reducing hallucinations by editing LLMs within a truthful space. Building on these works, we introduce a novel approach to improve LLM logical validity through representation modification. We show that LLM representations can be projected into content and logic spaces, and that modifying representations in the logic space can enhance or correct the logical validity of LLMs.

Method

Overview

LCF is a neural network that can be added after the attention and MLP modules of a transformer(Vaswani et al. 2017). It takes the raw output from attention or the MLP (i.e., the hidden representation) as input and produces a modified hidden representation as output. LCF is illustrated in Figure 2.

Content and Logic Projectors

Given a hidden representation R_{input}, i.e., the output of the self-attention or MLP module in the transformer. LCF maps R_{input} to both content and logic spaces using trained content and logic projectors, resulting in content representation $R_{content}$ and logic representation R_{logic}. Both the content and logic projectors are multi layer perceptron. To preserve the content of the representation, LCF only modifies R_{logic} while keeping $R_{content}$ unchanged. The modification of R_{logic} is similar to gradient updating, adjusting R_{logic} towards the logically valid region. The modification formula and the calculation formula for the logically valid direction are as follows:

$$R_{logic+} = R_{logic} + \mathbb{V}, \tag{1}$$

$$\mathbb{V} = C^{pos}_{logic} - C^{neg}_{logic}, \tag{2}$$

where $\mathbb{V}$ is the logical valid direction. C^{pos}_{logic} and C^{neg}_{logic} represent the central points of logically valid region and logically invalid region in the logic space, calculated as the mean of all the logically valid representation and all the invalid representation, respectively.

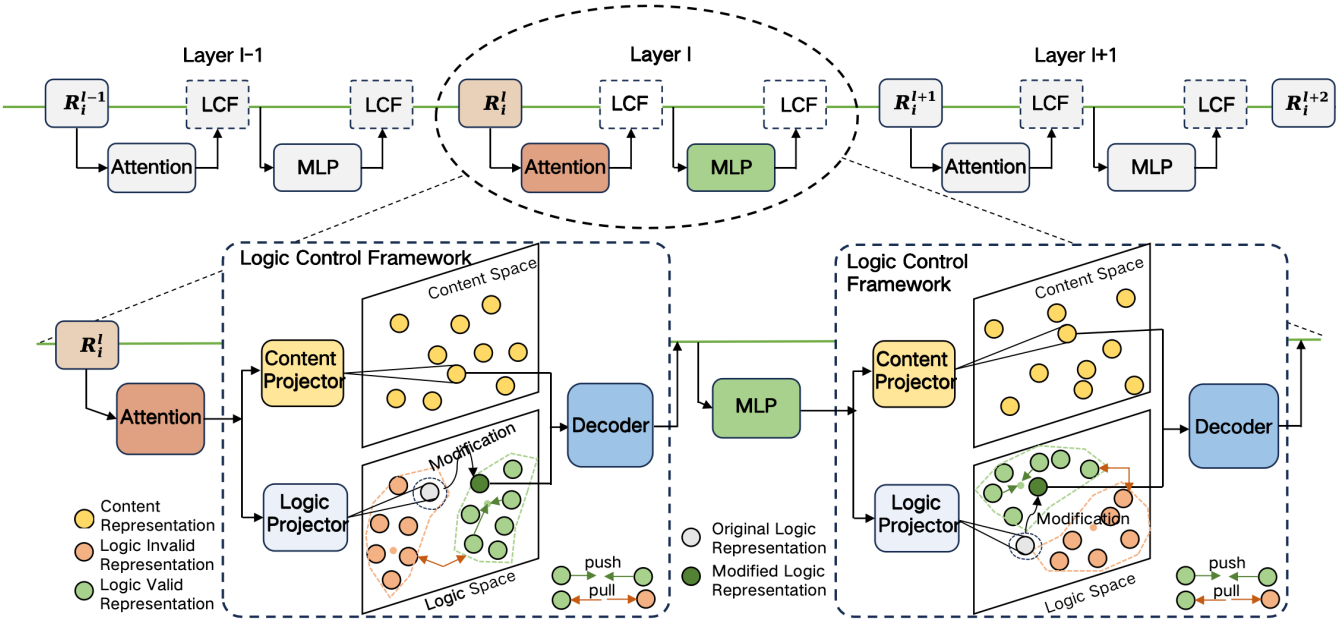

Figure 2: Logic Control Framework to modify the hidden representation in Transformers.

Representation Fusion

LCF contains a decoder to maps $R_{content}$ and the modified R_{logic+} back to the original representation space. The decoder is also a multilayer perceptron. Specifically:

$$\begin{aligned} R_+ &= Decoder(R_{content}, R_{logic+}), \\ &= MLP(R_{content} + Attn(R_{content}, R_{logic+})), \end{aligned} \tag{3}$$

where $Attn$ is a cross attention between $R_{content}$ and R_{logic+}. R_+ includes both the content representation and the modified logical representation. We use it as an anchor point to adjust the original hidden representation R_{input}. Specifically, we adjust R_{input} to move closer to R_+:

$$\mathbb{D} = R_+ - R_{input}, \tag{4}$$

$$R_{input+} = R_{input} + \frac{\mathbb{D}}{\|\mathbb{D}\|_2} \times \eta, \tag{5}$$

where $\frac{\mathbb{D}}{\|\mathbb{D}\|_2}$ is the modification direction and η represents the magnitude of the modification. R_{input+} will replace R_{input} as input to the next module of the Transformer.

Training

In LCF, both projectors and the decoder are trainable. We use a commonly used reconstruction objective to train these three modules:

$$\hat{R} = Decoder(R_{content}, R_{logic}), \tag{6}$$

$$\mathcal{L}_{rec} = MSE(R_{input}, \hat{R}). \tag{7}$$

Relying solely on reconstruction loss does not ensure that the content projector learns logic-independent representations or that the logic projector learns content-independent representations(Li, Cai, and Wu 2024). If there is overlap between these representations, modifying the logic representation may impact the content. To achieve more precise disentanglement, we introduce two additional constraints for the projectors.

Logic Projector Constraint. For the logic projector, we expect it to extract significantly different representations from logically valid and logically invalid samples. Therefore, we introduce contrastive learning (Wang and Isola 2020; Xu et al. 2023) to constrain the logic projector. Specifically, we minimize the distance among representations of logically valid samples, denoted as $S_{logic+} = \{R^1_{logic+}, R^2_{logic+}, ..., R^k_{logic+}\}$, where k is the dataset size. Simultaneously, we maximize the distance between each R^i_{logic+} and all representations of logically invalid samples, $S_{logic-} = \{R^1_{logic-}, R^2_{logic-}, ..., R^k_{logic-}\}$:

$$\mathcal{L}_{\text{logic+}} = \mathop{\mathbb{E}}_{(x,y)\sim S_{\text{logic+}}} \left[-\log \frac{e^{sim(x,y)/\tau}}{e^{sim(x,y)/\tau} + \sum_j^k e^{sim(R^j_{logic-},y)/\tau}} \right]. \tag{8}$$

Similarly, we minimize the distance between representations of logically invalid samples S_{logic-}, while maximize the distance between each R^i_{logic-} and all representations of logically invalid samples S_{logic+}:

$$\mathcal{L}_{\text{logic-}} = \mathop{\mathbb{E}}_{(x,y)\sim S_{\text{logic-}}} \left[-\log \frac{e^{sim(x,y)/\tau}}{e^{sim(x,y)/\tau} + \sum_j^k e^{sim(R^j_{logic+},y)/\tau}} \right]. \tag{9}$$

LLMs		Conclusion Generation			Fallacy Identification	
		Valid%(GPT4) ↑	Valid%(Trained) ↑	Perplexity ↓	Accuracy ↑	Δ Prob. ↑
Llama2	Original	70.58	58.84	21.08	51.47	-1.89
(Touvron et al. 2023)	+LCF	83.82	**96.56**	**12.12**	75.00	**6.29**
Llama3	Original	70.58	51.47	32.54	50.00	-0.90
(Dubey et al. 2024)	+LCF	82.84	93.13	17.76	**76.96**	5.12
Vicuna	Original	72.54	58.82	22.81	49.01	-1.21
(Chiang et al. 2023)	+LCF	78.92	75.00	20.39	71.56	4.71
Mistral	Original	80.88	67.64	30.53	57.84	-1.66
(Jiang et al. 2023)	+LCF	**85.71**	94.60	21.17	74.01	3.39
ChatGLM3	Original	74.50	52.94	96.95	47.54	-3.29
(GLM et al. 2024)	+LCF	77.94	93.62	42.47	73.03	3.02
Baichuan	Original	78.32	68.62	35.63	50.49	-3.26
(Yang et al. 2023)	+LCF	81.86	91.17	29.82	66.17	1.59

Table 1: LCF improves the logical validity of LLMs in both generation and discrimination tasks. The Valid%(GPT4) and Valid%(Trained) represent that the validity of generated conclusions are determined by GPT4 or a trained model, respectively.

Content Projector Constraint. For the content projector, we expect that the content representations it extracts are logically independent. Consequently, the same content representation should reconstruct into hidden representations with varying logical validity when combined with different logical representations. Specifically, for a pair of hidden representations R_{input+} and R_{input-} that share the same content but exhibit opposite logical validity, the constraint on the content projector is:

$$\hat{R}_{-} = Decoder(R_{content+}, R_{logic-}), \tag{10}$$

$$\hat{R}_{+} = Decoder(R_{content-}, R_{logic+}), \tag{11}$$

$$\mathcal{L}_{\text{content}} = MSE(R_{input-}, \hat{R}_{-}) + MSE(R_{input+}, \hat{R}_{+}). \tag{12}$$

Finally, we optimize the reconstruction loss and the constraints of the logic projector and content projector:

$$\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{logic+} + \mathcal{L}_{logic-} + \mathcal{L}_{content}. \tag{13}$$

Experimental Setup

Dataset

We use the LFUD dataset (Li et al. 2024b) for testing. LFUD is a logical fallacy understanding dataset featuring 12 types of fallacies across 67 scenarios, encompassing a total of 804 data points. We partition the dataset into training, validation, and test sets in a 45:5:17 ratio by scenario, yielding 540, 60, and 204 data points, respectively. The test set consists of scenarios that are distinct from those in the training and validation sets, enabling us to assess the proposed method's effectiveness in separating content and logic in unseen scenarios. We introduce two types of evaluation tasks: conclusion generation and fallacy identification. These tasks assess whether LCF can mitigate fallacies in LLMs from both generative and discriminative perspectives.

Conclusion Generation. Given several natural language premises, an LLM must generate a valid conclusion based on these premises. The evaluation metric, Valid%, is defined as the proportion of valid conclusions generated, calculated as the number of valid conclusions divided by the total number of conclusions generated. We use two discriminators to assess conclusion validity: (1) the GPT-4 Discriminator, which achieves an accuracy rate of 80% (Li et al. 2024b), though its judgments occasionally do not align with the fallacy types in LFUD; and (2) the Trained Llama-2 Discriminator, specifically trained to identify the 12 types of fallacies in LFUD, achieving over 90% accuracy in our manual verification. The metrics computed by these discriminators are referred to as **Valid%(GPT4)** and **Valid%(Trained)**, respectively. Additionally, **Perplexity** is used to measure the text generation capability of LLMs.

Fallacy Identification. Given several natural language premises, an LLM must select the valid conclusion from four options: one valid conclusion, two invalid conclusions, and "I have no comment." We employ two evaluation metrics: 1. **Accuracy**: the proportion of correct samples to the total number of samples in the test set. 2. Δ **Probability**: the average difference between the probability assigned to the correct answer and the probabilities assigned to the incorrect answers. This metric assesses the LLM's ability to distinguish between valid and invalid conclusions. A larger Δ Probability indicates a stronger ability of the LLM to identify fallacies.

Base LLMs

We select several widely used open-source LLMs to evaluate the effectiveness of LCF. Including Llama2 (Llama-2-7b-chat-hf (Touvron et al. 2023)), Llama3 (Meta-Llama-3.1-8B-Instruct (Dubey et al. 2024)), Vicuna (vicuna-7b-v1.5 (Chiang et al. 2023), Mistral (Mistral-7B-Instruct-v0.2) (Jiang et al. 2023)), ChatGLM3 (chatglm3-6b (GLM et al.

	Case 1 (Intentional fallacy)	**Case 2** (Ad hominem)
Premise	No one has ever seen Brenda not practicing the violin.	You're always complaining about something.
Llama2	Brenda is an excellent violinist.	You never seem to be satisfied with anything.
Llama2+LCF	Therefore, it is likely that Brenda is a skilled violinist.	Based on this statement, it may be helpful to identify the specific issue or issue(s) that are being complained about and to explore possible ways of addressing them.
	Case 3 (Ad populum)	**Case 4** (Deductive fallacy)
Premise	A lot of people think that the garden needs watering.	If some students do not take their studies seriously, failure is inevitable. Some students take their studies seriously.
Llama2	The garden needs watering.	Students who take their studies seriously will not fail.
Llama2+LCF	The garden may need watering, but it is not a certainty.	It is important to note that those who do take their studies seriously may still experience failure, as success is not solely dependent on academic effort.

Table 2: After the modification by LCF, the validity of the conclusions generated by LLMs is improved.

2024)), BaiChuan (Baichuan2-7B-Chat (Yang et al. 2023)).

Implementation Details

Training data. Training LCF involves using hidden representations with identical content but opposite logical validity (R_{input+} and R_{input-}). However, LFUD's training data includes only 540 fallacious samples and lacks logically valid counterparts. To address this, we use GPT-3.5-turbo to generate logically valid conclusions for each sample. We manually review and revise these generated samples to ensure their validity, resulting in 540 logically valid samples and 540 logically invalid samples. From these, we extract R_{input+} and R_{input-} pairs. Details on the specific data used for training each LLM's LCF are provided in the supplementary materials.

Training details. In LCF, both the content projector and the logic projector are two-layer MLPs with projection dimensions of 2048 and 1024. The decoder has dimensions of 1024 and 2048. We use AdamW (Loshchilov and Hutter 2017) to optimize for 10 epochs with a learning rate of 1e-3. The values of η are 0.5 and 4.5 for conclusion generation and fallacy identification tasks, respectively. LCF only modifies the 10 attention or MLP layers with the highest distinctiveness.

Results and Analysis

Overall Performance

Generation Ability. As shown in Table 1, the LCF method significantly enhances the validity of conclusions generated by LLMs. According to the Valid%(Trained) metric, LCF yields a 10%-40% improvement across all six tested LLMs. The most substantial increase is 41.66% for Llama3, while the smallest is 16.16% for Vicuna. After incorporating LCF, 96.56% of Llama2-generated conclusions are deemed valid, followed by Mistral at 94.60%. Moreover, all LLMs with LCF produced conclusions with at least 90% validity. These results demonstrate that LCF effectively improves the performance of existing transformer-based LLMs.

For the GPT-4 evaluation, LCF results in a 3%–13% improvement in validity. The Llama2+LCF combination provides the largest increase of 13.24% among all LLMs. In the GPT-4 evaluation, Mistral demonstrates the highest conclusion validity at 80.88%, which rises to 85.71% with LCF. This suggests that LCF positively enhances the logical validity of all LLMs. Additionally, perplexity metrics show that LCF does not negatively impact the inherent language modeling capabilities of LLMs; all LLMs exhibit comparable or reduced perplexity after incorporating LCF. We also conduct human evaluations: three annotators analyzed 50 randomly selected samples. The results show that all three annotators observed a clear improvement in the logical validity of LCF for LLMs. More anlysis and cases are shown in Supplementary Materials [1].

Identification Ablitiy. In the fallacy identification task, LLM+LCF significantly improves performance, with Llama3 showing the largest increase in accuracy at 26.96%, and Baichuan the smallest at 15.68%. This suggests that LCF enhances not only the generation capabilities of LLMs but also their ability to identify fallacies. Through LCF modifications, LLMs' hidden representations tend to produce logically valid outputs, thereby increasing the probability of such outputs. This enhancement helps LLMs better distinguish between valid and fallacious conclusions, as confirmed by the Δ Probability metric. Prior to LCF incorporation, LLMs' Δ Probability was negative, indicating a poor distinction between valid and invalid conclusions, with a higher probability of invalid conclusions. After LCF incorporation, all LLMs showed improved discernment, with Δ Probability values becoming positive, and Llama2 demonstrated the highest increase at 6.29%. This indicates that contrastive learning, which amplifies the gap between logically valid and invalid representations, enhances LLMs' ability to differentiate valid conclusions from fallacies.

[1] https://github.com/wulidongdong/LCF

	Valid%	Accuracy	Δ Prob.
Llama2	57.84	51.47	-1.89
Llama2+LCF	96.56	75.00	6.29
w/o L_{rec}	93.13	76.96	6.95
w/o $L_{content}$	94.60	75.00	5.46
w/o L_{logic}	93.62	51.47	-1.83
w/o Content Proj.	82.84	73.52	5.64

Table 3: Ablation studies of LCF.

Case Study

We show examples in Table 2 of how modifying LLMs with LCF improves their logical validity. Overall, LCF enhances the objectivity of LLMs' responses and the comprehensiveness of their problem analysis. For instance, in Case 1 and Case 4, the responses generated by LLMs+LCF are less absolute and retain the possibility of other factors being involved. Additionally, LCF reduces the risk of LLMs echoing others and potential personal attacks. For example, in Case 3, LLMs+LCF do not assume that a viewpoint is correct just because it is held by the majority. In Case 2, LLMs+LCF also focus more on the events themselves rather than evaluating people. More cases are in the supplementary materials.

Ablation Study

Ablation results are shown in Table 3. Removing the content projector from LCF (without content projector) results in decreases (13.72% and 0.65%)in both Valid% and Accuracy metrics, but still higher than original Llama2. This suggests that modifying the hidden representations of LLMs directly can enhance logical validity, but it is susceptible to content interference. This finding supports the effectiveness of separating content from logic. Additionally, ablation experiments on the three types of loss used to train LCF reveal that individually removing $\mathcal{L}content$, $\mathcal{L}logic$, and $\mathcal{L}_{rec}$ reduces the Valid% of conclusions generated by LLMs to 93.13%, 94.60%, and 93.62%, respectively.

In contrast, the performance on discrimination tasks, such as accuracy and Δ Probability, significantly declines when $\mathcal{L}_{logic}$ is omitted. Accuracy drops from 75.00% to 51.47%, and Δ Probability decreases from 6.29% to -1.83%. This shows that contrastive learning significantly helps LCF learn to distinguish between logically valid and invalid content.

Visualization

We conduct a visual analysis of the content and logic spaces in LCF. Using the t-SNE (Van der Maaten and Hinton 2008) method, we reduce the dimensionality of validation set samples to 2D. Figure 3 (a) illustrates the distribution of logically valid and invalid samples in both spaces after processing through the content and logic projectors. The content space exhibits a uniform distribution due to LCF's lack of control over content representation, while the logic space reveals a clear boundary between valid and invalid logical representations. This separation suggests that contrastive learning effectively distinguishes between valid and invalid representations in the logic space. Additionally, Figures 3(b) and (c) display eight different logic spaces across layers. The first four layers (Figure 3(b)) show logic spaces with higher discriminative power, where LCF can clearly differentiate between valid and invalid representations. In contrast, the latter four layers (Figure 3(c)) present logic spaces with lower discriminative power, where valid and invalid representations are intermixed with blurred boundaries. This indicates that not all hidden representations across layers are equally suitable for modification.

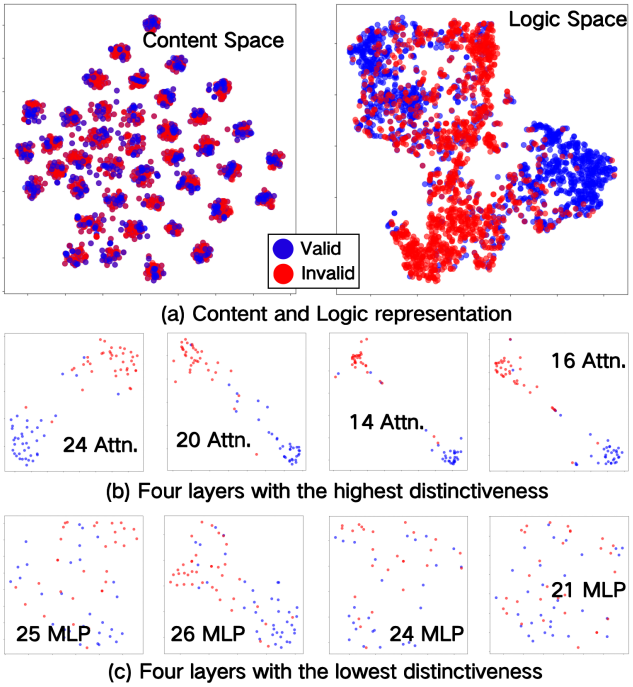

Figure 3: Logic Control Framework to modify the hidden representation in Transformers.

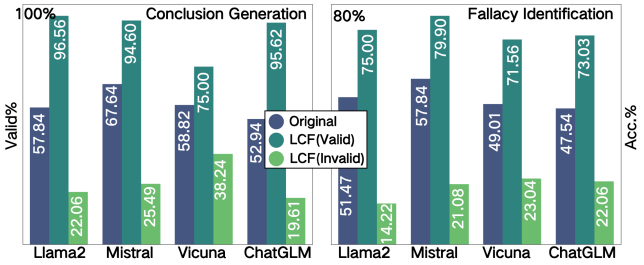

Figure 4: Comparison between LLMs with valid and invalid modification using LCF.

Invalid Modification

In addition to enhancing the logical validity of LLMs, LCF can also be used to reduce it. By reversing the direction in Formula (4), we modify the hidden representations to move away from the logically valid region. Figure 4 and Table 4 illustrate the performance comparison and examples following invalid modifications. After applying invalid modifications to the tested LLMs, there is a marked decline in the va-

	Case1
Premise	Gardeners claim that all flowers don't stay open forever. Gardeners are experts in the field concerning flowers.
+LCF-V	While gardeners may have extensive knowledge about flowers and their behavior, their claim that all flowers don't stay open forever may not be universally true.
+LCF-I	Flowers don't stay open forever.
	Case2
Premise	Anna claims she finished the report on time, but we all know she took a nap in the afternoon. She's a slacker.
+LCF-V	Anna's claim that she finished the report on time may not be entirely accurate due to her afternoon nap.
+LCF-I	Anna is not trustworthy.

Table 4: Generated conclusions of LLMs with valid (LCF-V) and invalid (LCF-I) modification.

lidity of the generated conclusions. The Valid% metric drops by 20.58%-42.15% compared to the original LLMs and by 36.76%-74.50% compared to LLMs+LCF (Valid). For fallacy detection, accuracy decreases by 25.48%-37.25% compared to the original LLMs and by 48.52%-60.78% compared to LLMs+LCF (Valid). As shown in Table 4, conclusions generated by LLMs after LCF invalid modification exhibit strong absoluteness, a characteristic typical of conclusions with logical fallacies. These results demonstrate that LCF can alter logical validity in both directions.

Modification Layer Analysis

Based on the visualization results, the decoupling and modification capabilities of the LCF vary across different layers of hidden representations. We limit modifications to one layer at a time and analyze their impact on the results. Specifically, we examine how modifying layers 10-30 affects conclusion generation and fallacy identification. Figure 5 (top) presents the Valid% of conclusion generation for modifications to individual attention (a) and MLP (b) layers. Figure 5 (bottom) shows the Accuracy of fallacy identification for the same modifications (c and d). The results reveal a clear pattern: modifying attention layers between 10 and 20 significantly improves both conclusion generation and fallacy identification, whereas modifications to attention layers beyond 20 show diminished performance. In contrast, modifying MLP layers between 20 and 30 yields better results compared to modifying MLP layers between 10 and 20. Additionally, the optimal number of layers to modify is task-dependent, but overall, modifications to attention and MLP layers in the 15-20 range generally enhance performance.

Supervised Finetuning Comparison

We conduct comparison between two truthfulness-oriented baselines, ITI (Li et al. 2024a) and RAHF (Liu et al. 2024b).

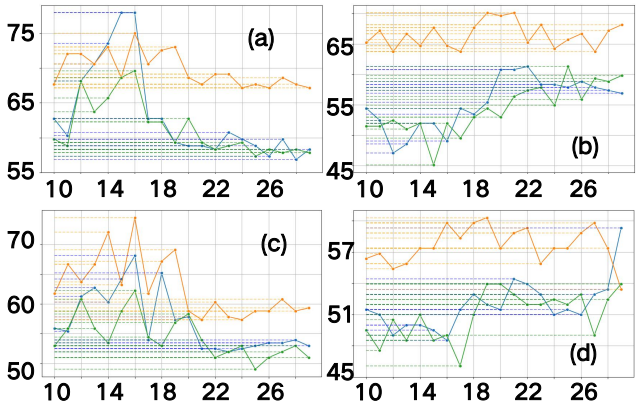

Figure 5: Modifying LLMs of single attention or MLP layer. (a) Attention Layer on Conclusion Generation (Valid%). (b) MLP Layer on Conclusion Generation (Valid%). (c) Attention Layer on Fallacy Identification (Acc%). (d) MLP Layer on Fallacy Identification (Acc%). Llama2(blue), Mistral(orange), Vicuna(green).

	Valid%(GPT-4)	Valid%(Trained)
Llama2	70.58	58.84
ITI	69.60	62.25
RAHF	71.56	46.56
SFT	79.90	78.43
LCF(Ours)	**83.82**	**96.56**

Table 5: Comparison with Supervised Fine-tuning.

Results in Table 5 suggest that these methods designed for improving truthfulness have limited improvement in logical validity of LLMs. On the contrary, supervised fine-tuning is a more straightforward and powerful baseline. After SFT training on Llama2, logical validity improves by 20%, though this enhancement is not as substantial as with LCF (38%). These findings demonstrate the effectiveness of the proposed LCF method.

Conclusion

We propose a logic control framework (LCF) designed to decouple and modify the hidden representations of LLMs. LCF projects the attention and MLP outputs of LLMs into separate content and logic spaces. By adjusting representations in the logic space to align with regions of logically valid representations, LCF significantly enhances the logical validity of LLMs. Evaluations across six LLMs in tasks such as conclusion generation and fallacy identification demonstrate LCF's effectiveness and robustness. We also quantitatively assess the impact of factors like modification magnitude and layer count. Furthermore, case studies and visualizations confirm that LCF substantially improves the validity of conclusions generated by LLMs and their ability to identify fallacies.

Acknowledgements

This research is supported by the Science and Technology Planning Project of Guangdong Province (2020B0101100002), the National Natural Science Foundation of China (62076100, 62476097), the Fundamental Research Funds for the Central Universities, South China University of Technology (x2rjD2240100), Guangdong Provincial Fund for Basic and Applied Basic Research—Regional Joint Fund Project (Key Project) (2023B1515120078), Guangdong Provincial Natural Science Foundation for Outstanding Youth Team Project (2024B1515040010), the China Computer Federation (CCF)-Zhipu AI Large Model Fund.

References

Bregant, J. 2014. Critical thinking in education: why to avoid logical fallacies? *Problems of Education in the 21st Century*, 61(1): 18–27.

Bu, Y.; Wu, X.; Cai, Y.; Liu, Q.; Wang, T.; and Huang, Q. 2024. Error-Aware Generative Reasoning for Zero-Shot Visual Grounding. *IEEE Transactions on Multimedia*.

Cheng, P. W.; and Holyoak, K. J. 1985. Pragmatic reasoning schemas. *Cognitive psychology*, 17(4): 391–416.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; Stoica, I.; and Xing, E. P. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.

Dasgupta, I.; Lampinen, A. K.; Chan, S. C. Y.; Sheahan, H. R.; Creswell, A.; Kumaran, D.; McClelland, J. L.; and Hill, F. 2024. Language models show human-like content effects on reasoning tasks. arXiv:2207.07051.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. 2024. The Llama 3 Herd of Models. arXiv:2407.21783.

GLM, T.; Zeng, A.; Xu, B.; Wang, B.; Zhang, C.; Yin, D.; Rojas, D.; Feng, G.; Zhao, H.; Lai, H.; Yu, H.; Wang, H.; Sun, J.; Zhang, J.; Cheng, J.; Gui, J.; Tang, J.; Zhang, J.; Li, J.; Zhao, L.; Wu, L.; Zhong, L.; Liu, M.; Huang, M.; Zhang, P.; Zheng, Q.; Lu, R.; Duan, S.; Zhang, S.; Cao, S.; Yang, S.; Tam, W. L.; Zhao, W.; Liu, X.; Xia, X.; Zhang, X.; Gu, X.; Lv, X.; Liu, X.; Liu, X.; Yang, X.; Song, X.; Zhang, X.; An, Y.; Xu, Y.; Niu, Y.; Yang, Y.; Li, Y.; Bai, Y.; Dong, Y.; Qi, Z.; Wang, Z.; Yang, Z.; Du, Z.; Hou, Z.; and Wang, Z. 2024. ChatGLM: A Family of Large Language Models from GLM-130B to GLM-4 All Tools. arXiv:2406.12793.

Goyal, S.; Rastogi, E.; Rajagopal, S. P.; Yuan, D.; Zhao, F.; Chintagunta, J.; Naik, G.; and Ward, J. 2024. Healai: A healthcare llm for effective medical documentation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, 1167–1168.

Haita-Falah, C. 2017. Sunk-cost fallacy and cognitive ability in individual decision-making. *Journal of Economic Psychology*, 58: 44–59.

Jason, G. 1989. Fallacies are common. *Informal Logic*, 11(2).

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; de las Casas, D.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; Lavaud, L. R.; Lachaux, M.-A.; Stock, P.; Scao, T. L.; Lavril, T.; Wang, T.; Lacroix, T.; and Sayed, W. E. 2023. Mistral 7B. arXiv:2310.06825.

Jin, Z.; Lalwani, A.; Vaidhya, T.; Shen, X.; Ding, Y.; Lyu, Z.; Sachan, M.; Mihalcea, R.; and Schoelkopf, B. 2022. Logical Fallacy Detection. In Goldberg, Y.; Kozareva, Z.; and Zhang, Y., eds., *Findings of the Association for Computational Linguistics: EMNLP 2022*, 7180–7198. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.

Kong, L.; Wang, H.; Mu, W.; Du, Y.; Zhuang, Y.; Zhou, Y.; Song, Y.; Zhang, R.; Wang, K.; and Zhang, C. 2024. Aligning Large Language Models with Representation Editing: A Control Perspective. arXiv:2406.05954.

Li, K.; Patel, O.; Viégas, F.; Pfister, H.; and Wattenberg, M. 2024a. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36.

Li, L.; Cai, Y.; and Wu, X. 2024. Unsupervised Disentanglement Learning Model for Exemplar-Guided Paraphrase Generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Li, Y.; Wang, D.; Liang, J.; Jiang, G.; He, Q.; Xiao, Y.; and Yang, D. 2024b. Reason from Fallacy: Enhancing Large Language Models' Logical Reasoning through Logical Fallacy Understanding. In *Findings of the Association for Computational Linguistics: NAACL 2024*, 3053–3066.

Li, Y.; Zhou, Q.; Luo, Y.; Ma, S.; Li, Y.; Zheng, H.-T.; Hu, X.; and Yu, P. S. 2024c. When LLMs Meet Cunning Texts: A Fallacy Understanding Benchmark for Large Language Models. arXiv:2402.11100.

Lim, G.; and Perrault, S. T. 2024. Evaluation of an LLM in Identifying Logical Fallacies: A Call for Rigor When Adopting LLMs in HCI Research. arXiv:2404.05213.

Liu, H.; Ning, R.; Teng, Z.; Liu, J.; Zhou, Q.; and Zhang, Y. 2023. Evaluating the Logical Reasoning Ability of ChatGPT and GPT-4. arXiv:2304.03439.

Liu, S.; Ye, H.; Xing, L.; and Zou, J. Y. 2024a. In-context Vectors: Making In Context Learning More Effective and Controllable Through Latent Space Steering. In *Forty-first International Conference on Machine Learning*.

Liu, W.; Wang, X.; Wu, M.; Li, T.; Lv, C.; Ling, Z.; Zhu, J.; Zhang, C.; Zheng, X.; and Huang, X. 2024b. Aligning Large Language Models with Human Preferences through Representation Engineering. arXiv:2312.15997.

Loshchilov, I.; and Hutter, F. 2017. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

Monti, M. M.; and Osherson, D. N. 2012. Logic, language and the brain. *Brain research*, 1428: 33–42.

Musi, E.; and Reed, C. 2022. From fallacies to semi-fake news: Improving the identification of misinformation triggers across digital media. *Discourse & Society*, 33(3): 349–370.

Naveed, H.; Khan, A. U.; Qiu, S.; Saqib, M.; Anwar, S.; Usman, M.; Akhtar, N.; Barnes, N.; and Mian, A. 2024. A Comprehensive Overview of Large Language Models. arXiv:2307.06435.

Payandeh, A.; Pluth, D.; Hosier, J.; Xiao, X.; and Gurbani, V. K. 2024. How Susceptible Are LLMs to Logical Fallacies? In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 8276–8286.

Sourati, Z.; Ilievski, F.; Sandlin, H.-n.; and Mermoud, A. 2023. Case-Based Reasoning with Language Models for Classification of Logical Fallacies. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Sun, Z.; Zhang, K.; Yu, W.; Wang, H.; and Xu, J. 2024. Logic Rules as Explanations for Legal Case Retrieval. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 10747–10759.

Tindale, C. W. 2007. *Fallacies and argument appraisal*. Cambridge University Press.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; Bikel, D.; Blecher, L.; Ferrer, C. C.; Chen, M.; Cucurull, G.; Esiobu, D.; Fernandes, J.; Fu, J.; Fu, W.; Fuller, B.; Gao, C.; Goswami, V.; Goyal, N.; Hartshorn, A.; Hosseini, S.; Hou, R.; Inan, H.; Kardas, M.; Kerkez, V.; Khabsa, M.; Kloumann, I.; Korenev, A.; Koura, P. S.; Lachaux, M.-A.; Lavril, T.; Lee, J.; Liskovich, D.; Lu, Y.; Mao, Y.; Martinet, X.; Mihaylov, T.; Mishra, P.; Molybog, I.; Nie, Y.; Poulton, A.; Reizenstein, J.; Rungta, R.; Saladi, K.; Schelten, A.; Silva, R.; Smith, E. M.; Subramanian, R.; Tan, X. E.; Tang, B.; Taylor, R.; Williams, A.; Kuan, J. X.; Xu, P.; Yan, Z.; Zarov, I.; Zhang, Y.; Fan, A.; Kambadur, M.; Narang, S.; Rodriguez, A.; Stojnic, R.; Edunov, S.; and Scialom, T. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In Guyon, I.; von Luxburg, U.; Bengio, S.; Wallach, H. M.; Fergus, R.; Vishwanathan, S. V. N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 5998–6008.

Wang, T.; and Isola, P. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, 9929–9939. PMLR.

Wason, P. C.; and Johnson-Laird, P. N. 1972. *Psychology of reasoning: Structure and content*, volume 86. Harvard University Press.

Xu, J.; Zheng, C.; Cai, Y.; and Chua, T.-S. 2023. Improving Named Entity Recognition via Bridge-based Domain Adaptation. In *Findings of the Association for Computational Linguistics: ACL 2023*, 3869–3882.

Yang, A.; Xiao, B.; Wang, B.; Zhang, B.; Bian, C.; Yin, C.; Lv, C.; Pan, D.; Wang, D.; Yan, D.; Yang, F.; Deng, F.; Wang, F.; Liu, F.; Ai, G.; Dong, G.; Zhao, H.; Xu, H.; Sun, H.; Zhang, H.; Liu, H.; Ji, J.; Xie, J.; Dai, J.; Fang, K.; Su, L.; Song, L.; Liu, L.; Ru, L.; Ma, L.; Wang, M.; Liu, M.; Lin, M.; Nie, N.; Guo, P.; Sun, R.; Zhang, T.; Li, T.; Li, T.; Cheng, W.; Chen, W.; Zeng, X.; Wang, X.; Chen, X.; Men, X.; Yu, X.; Pan, X.; Shen, Y.; Wang, Y.; Li, Y.; Jiang, Y.; Gao, Y.; Zhang, Y.; Zhou, Z.; and Wu, Z. 2023. Baichuan 2: Open Large-scale Language Models. arXiv:2309.10305.

yuan, l.; Cai, Y.; and Huang, J. 2024. Few-Shot Joint Multimodal Entity-Relation Extraction via Knowledge-Enhanced Cross-modal Prompt Model. In *Proceedings of the 32nd ACM International Conference on Multimedia*, MM '24, 8701–8710. New York, NY, USA: Association for Computing Machinery. ISBN 9798400706868.

Zhang, S.; Yu, T.; and Feng, Y. 2024. TruthX: Alleviating Hallucinations by Editing Large Language Models in Truthful Space. arXiv:2402.17811.