CAUSAL INTERVENTION ON LATENT CONFOUNDING BIAS NEURONS IN LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Pre-training corpora of large language models (LLMs) often contain latent confounding information, which can be encoded into model parameters and induce confounder-related neuronal activations during inference. Existing causal debiasing methods for LLMs predominantly rely on the front-door adjustment criterion, which assumes that latent confounders do not directly affect the reasoning process. In practice, however, latent confounders may also affect the reasoning process, violating the assumptions required for valid front-door adjustment. To address this limitation, we propose CILC (Causal Intervention on Latent Confounding Bias Neurons), a neuron-level causal intervention framework for mitigating latent confounding bias in LLMs. Specifically, CILC first employs a Variational Autoencoder (VAE) to infer latent representations from the data and maps these representations into the LLM's input embedding space via a latent-to-soft-prompt transformation. It then constructs contrastive inputs with and without the resulting latent soft prompts and applies Backward Integrated Gradients (BIG) to localize neurons that are particularly sensitive to latent confounding information. Finally, CILC performs targeted causal interventions on the identified neurons to attenuate the influence of latent confounding on model predictions. Extensive experiments on three LLM backbones and eight real-world datasets demonstrate that CILC effectively mitigates latent confounding bias while consistently improving model performance. Our source code is available at `https://anonymous.4open.science/r/CILC-3E63`.

1 INTRODUCTION

Large Language Models (LLMs) (Ouyang et al., 2022; Brown et al., 2020; Touvron et al., 2023) have achieved remarkable progress across a wide range of applications, including question answering (Gozalo-Brizuela & Garrido-Merchan, 2023; Yao et al., 2023), dialogue (Ouyang et al., 2022; Thoppilan et al., 2022), and recommendation systems (Bao et al., 2023; Zhang et al., 2025b). Much of this progress can be attributed to large-scale pretraining on massive internet corpora. However, these corpora inevitably contain information associated with social backgrounds, user preferences, behavioral patterns, and complex statistical dependencies among them. Even when the underlying confounding factors are not directly observed, information associated with them may be implicitly learned and encoded in model parameters during pretraining. Consequently, during inference, internal representations associated with latent confounding factors may be activated and influence model predictions, potentially giving rise to latent confounding bias (Nadeem et al., 2021; Liang et al., 2021; Parrish et al., 2022).

Existing approaches on LLM debiasing have primarily focused on biases associated with observed attributes or confounding factors (Nadeem et al., 2021; Navigli et al., 2023; Li et al., 2025; Oba et al., 2024; Guo et al., 2022). For example, Prompting Fairness (Li et al., 2025) leverages causal selection mechanisms to mitigate model bias; ICGBS (Oba et al., 2024) reduces gender bias through manually designed counterfactual prompts; and Auto-Debias (Guo et al., 2022) employs distribution alignment objectives to mitigate biases associated with demographic attributes. Despite their effectiveness, these methods generally require the relevant biased attributes to be explicitly observed or specified, limiting their ability to address biases arising from unobserved confounding factors.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Addressing latent confounding bias introduces an additional challenge: latent confounders cannot be directly observed, yet information associated with them may have already been encoded in LLM parameters during pretraining (Cheng et al., 2022; 2024). During inference, neurons that encode or respond to such information may become activated and consequently influence model predictions (Pearl, 2009). Representation learning methods such as Variational Autoencoders (VAEs) (Kingma & Welling, 2013) provide a natural means of extracting latent factors from observable data. However, the resulting latent representations typically reside in a continuous latent space that is disconnected from the LLM's input embedding space. As a result, these representations cannot be directly introduced into the LLM to characterize their effects on internal activations and model predictions. Establishing an actionable connection between inferred latent representations and the LLM input space is therefore a key challenge for identifying and mitigating latent confounding bias.

Recent studies have explored causal inference as a means of mitigating latent bias in LLMs. For example, Causal Prompting (Zhang et al., 2025a) treats Chain-of-Thought (CoT) reasoning (Wei et al., 2022) as a mediator and estimates causal effects through front-door adjustment. ACPS (Li et al., 2026) further combines adaptive front-door adjustment with lightweight inference to improve inference efficiency. However, the validity of front-door adjustment depends on assumptions concerning the relationships among the treatment, mediator, outcome, and unobserved confounders. In particular, the treatment–mediator relationship must not be subject to unblocked confounding. In the context of LLMs, information associated with latent confounders may already be encoded in model parameters and may therefore affect the intermediate reasoning process used as the mediator. When such latent factors jointly influence the relevant causal variables and the mediator, the assumptions required for front-door identification may no longer hold. Moreover, these approaches depend on an explicit intermediate reasoning process, such as CoT, whereas many practical LLM applications do not require or expose explicit reasoning traces. This dependence further limits the applicability of front-door-based debiasing methods.

To address these limitations, we propose CILC (Causal Intervention on Latent Confounding Bias Neurons), a neuron-level causal intervention framework for mitigating latent confounding bias in LLMs. Specifically, CILC first employs a VAE to infer latent representations that capture potential confounding information from observed data and maps these representations into the LLM input embedding space through a latent-to-soft-prompt mapping module. This mapping enables latent information to be explicitly introduced into the LLM, providing a means to examine how it affects internal neuronal activations. CILC then constructs contrastive inputs with and without the resulting latent soft prompts and applies Backward Integrated Gradients (BIG) to localize neurons that are particularly sensitive to latent confounding information. Finally, CILC performs targeted causal interventions on the activations of the identified neurons during inference, thereby attenuating the influence of latent confounding information on model predictions.

In summary, our main contributions are as follows:

- We design a latent-to-soft-prompt mapping mechanism that maps VAE-inferred latent representations into the LLM input embedding space. This mechanism establishes an actionable connection between latent representations and the LLM, enabling their effects on internal neuronal activations to be explicitly characterized.
- We propose CILC, a neuron-level causal intervention framework that localizes neurons sensitive to latent confounding information through contrastive inputs and BIG, and performs targeted interventions on their activations. Unlike front-door-based approaches, CILC itself does not rely on an explicit reasoning process as a mediator, making its core neuron-level intervention applicable to a broader range of LLM inference settings.
- Extensive experiments across three LLM backbones and eight real-world datasets demonstrate that CILC effectively mitigates latent confounding bias while consistently improving predictive performance.

2 PRELIMINARIES

2.1 NOTATIONS

We first introduce the key notations used throughout this paper. Let X denote the input to the LLM, constructed from the observable user historical interactions, and let Y denote the corresponding

model output. We use Z to denote the latent confounder. To localize the neurons associated with Z, we infer a proxy representation $\widehat{Z}$ from X using a VAE.

To characterize the influence of the inferred latent information on the LLM, we construct two contrastive input embeddings. Specifically, $\mathbf{E}^{+Z}$ denotes the input embedding augmented with the proxy representation $\widehat{Z}$, whereas $\mathbf{E}^{-Z}$ denotes the corresponding input embedding without $\widehat{Z}$. By contrasting the neuronal responses induced by $\mathbf{E}^{+Z}$ and $\mathbf{E}^{-Z}$, we localize neurons that are sensitive to the latent confounding information represented by $\widehat{Z}$. We use H_Z to denote the activations of these localized latent-confounding-sensitive neurons.

2.2 Problem Statement

As shown in Fig. 1(a), we formalize the prediction process of LLMs using a structural causal model (SCM) (Pearl, 2009). Since LLMs encode latent confounders into the parameter space during pre-training, they are able to activate the corresponding neurons at inference time. We assume that the latent confounder Z affects these neuronal activations, i.e., $Z \rightarrow H_Z$. Therefore, the latent confounder forms a backdoor path $X \leftarrow Z \rightarrow H_Z \rightarrow Y$, through which latent confounding information can influence the model output Y. Consequently, estimating the effect of X on Y without accounting for this path may yield predictions contaminated by latent confounding bias.

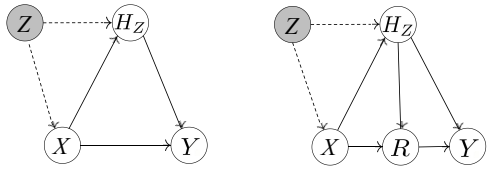


Figure 1: DAGs illustrating latent confounders in LLMs. (a) The proposed SCM without an explicit reasoning process. (b) The SCM incorporating an explicit reasoning process R, where R represents CoT or SoT reasoning. Z represents latent confounders, X and Y denote the model input and output, respectively, and H_Z denotes the activation of the bias-related neurons.

Existing causal debiasing methods typically treat the reasoning process R generated by CoT or SoT as a mediator and use the front-door adjustment criterion to estimate the causal effect of the input X on the output Y. As shown in Fig. 1(b), front-door adjustment requires that the mediator R can effectively block the backdoor path between X and Y. Since the latent confounder Z can influence the reasoning process through $Z \rightarrow H_Z \rightarrow R$ and form the backdoor path $X \leftarrow Z \rightarrow H_Z \rightarrow R \rightarrow Y$, the key condition required by front-door adjustment is violated, rendering the front-door adjustment criterion inapplicable.

Therefore, CILC identifies the neurons associated with the latent confounder and performs targeted interventions on their activations during inference, thereby mitigating the influence of the latent confounding bias propagated through the backdoor path between X and Y.

2.3 Theoretical Motivation

Standard front-door adjustment requires, among other conditions, that the mediator R is not affected by the latent confounder Z. However, in LLMs, information associated with the latent confounder Z may be encoded in model parameters during pretraining and subsequently influence the reasoning process through the internal activation pathway $Z \rightarrow H_Z \rightarrow R$. Together with $Z \rightarrow X$, this induces the backdoor path $X \leftarrow Z \rightarrow H_Z \rightarrow R$, thereby violating a key condition required for front-door identification. To formalize the causal justification of CILC, we establish the following proposition; the assumptions and the proof is deferred to Appendix B.

Proposition 1 (Attenuation of the latent-confounder contribution)**.** *Suppose Assumptions 1–4 hold. Let*

$$ATE = \mathbb{E}\big[Y \mid do(\mathbf{E} = \mathbf{E}^{+Z})\big] - \mathbb{E}\big[Y \mid do(\mathbf{E} = \mathbf{E}^{-Z})\big]$$

denote the output effect induced by introducing the latent-confounder proxy. Consider the neuron intervention

$$do\big(H_Z = \gamma H_Z\big), \qquad 0 \leq \gamma \leq 1.$$

Under the local intervention and local linear response assumptions, the first-order contribution of the identified latent-confounder pathway is attenuated approximately by a factor of γ. Therefore,

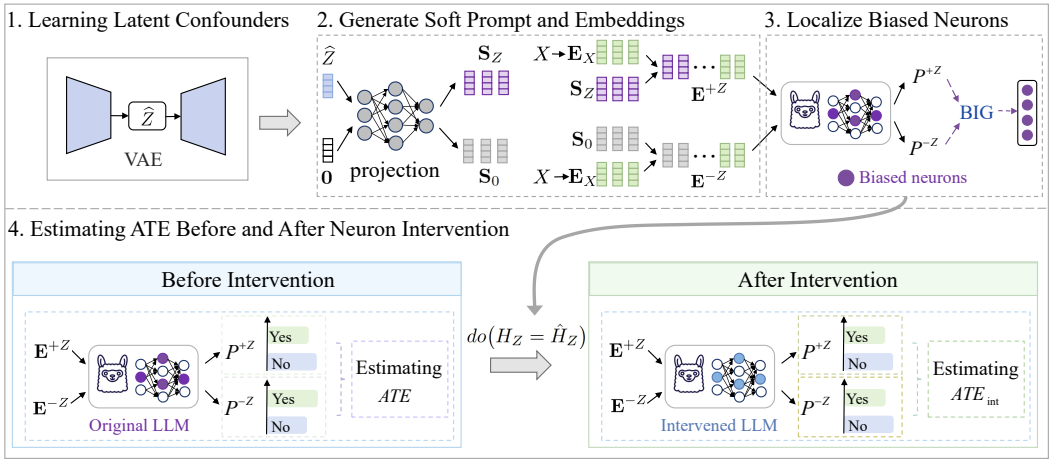


Figure 2: Overview of the proposed CILC framework. CILC first uses a VAE to learn latent representations and maps them into soft prompts acceptable to the LLM. Subsequently, by constructing inputs with and without the latent soft prompts, it uses backward integrated gradients (BIG) to localize bias-related neurons. Finally, CILC performs causal interventions on the activations of the bias-related neurons and estimates their causal effects on model predictions, thereby suppressing latent confounding bias.

when the measured effect is dominated by this identified pathway, the post-intervention effect admits the local approximation

$$ATE_{\text{int}} \approx \gamma ATE.$$

Accordingly, under the local linear approximation where non-targeted effects are treated as higher-order residual terms,

$$|ATE_{\text{int}}| \approx \gamma |ATE|,$$

and the reduction in the magnitude of the proxy-induced effect satisfies

$$\Delta ATE = |ATE| - |ATE_{\text{int}}| \approx (1-\gamma)|ATE|.$$

Thus, decreasing γ is expected to progressively attenuate the contribution of the identified latent-confounder pathway.

3 The Proposed CILC Method

In this section, we introduce the proposed CILC framework. As shown in Figure 2, CILC consists of three main stages: (1) using a VAE to infer latent representations and mapping them into soft prompts acceptable to the LLM, (2) using BIG to localize neurons associated with latent bias, and (3) performing causal intervention on the activations of the bias-related neurons and causal effect estimation. We describe each stage in detail in the following subsections.

3.1 Learning Latent Confounding Representations

Since the latent confounder Z is unobserved, we employ a VAE to infer a proxy representation $\widehat{Z}$ from the observed user history X, aiming to capture potential latent confounding information. Specifically, the VAE is trained by maximizing the evidence lower bound (ELBO):

$$\mathcal{L}(\theta, \phi) = \mathbb{E}_{q_\phi(\widehat{Z}|X)}[\log p_\theta(X|\widehat{Z})] - KL(q_\phi(\widehat{Z}|X) \| p_\theta(\widehat{Z})), \tag{1}$$

where $q_\phi(\widehat{Z}|X) = \mathcal{N}(\mu_\phi(X), \sigma^2_\phi(X))$ denotes the variational posterior parameterized by the encoder, $p_\theta(X \mid \widehat{Z})$ denotes the likelihood parameterized by the decoder, and $p(\widehat{Z}) = \mathcal{N}(0, I)$ denotes the standard Gaussian prior. Detailed derivations are deferred to Appendix C.1.

3.2 GENERATE SOFT PROMPT AND EMBEDDINGS

To establish an actionable connection between the inferred latent representation $\widehat{Z}$ and the LLM input space, we employ an MLP g_ψ to map $\widehat{Z}$ into B continuous soft prompt tokens. Specifically, we define:

$$\mathbf{S}_Z = \text{reshape}(g_\psi(\widehat{Z}), B, d_h), \quad \mathbf{S}_0 = \text{reshape}(g_\psi(\mathbf{0}), B, d_h).$$

Given an input prompt X, we obtain its token embeddings as:

$$\mathbf{E}_X = \text{Embed}(X), \tag{2}$$

where $\mathbf{E}_X \in \mathbb{R}^{L \times d_h}$, and L denotes the number of text tokens.

We then construct a pair of contrastive input representations by concatenating the corresponding soft prompts with the token embeddings:

$$\mathbf{E}^{+Z} = [\mathbf{S}_Z; \mathbf{E}_X], \quad \mathbf{E}^{-Z} = [\mathbf{S}_0; \mathbf{E}_X], \tag{3}$$

where $[\cdot;\cdot]$ denotes concatenation along the sequence dimension.

3.3 USE BIG TO LOCALIZE BIAS-RELATED NEURONS

To identify the bias-related neurons that contribute to the prediction differences induced by the latent representation $\widehat{Z}$, CILC employs Backward Integrated Gradients (BIG) (Sundararajan et al., 2017; Lin et al., 2026) to quantify the contribution of each individual neuron to the prediction difference between inputs with and without $\widehat{Z}$.

For an embedding input E_i^{+Z} that contains the latent representation, the corresponding hidden neuron activations are denoted as:

$$H_i^{+Z} = \{h_{i,1}^{+Z}, h_{i,2}^{+Z}, \cdots, h_{i,T}^{+Z}\}, \tag{4}$$

where $h_{i,t}^{+Z}$ represents the hidden activation vector corresponding to the t-th position in the input E_i^{+Z}, and T denotes the length of the input embedding sequence.

Similarly, for an embedding input E_i^{-Z} without the latent representation, we obtain:

$$H_i^{-Z} = \{h_{i,1}^{-Z}, h_{i,2}^{-Z}, \cdots, h_{i,T}^{-Z}\}, \tag{5}$$

where $h_{i,t}^{-Z}$ represents the hidden activation vector corresponding to the t-th position in the input E_i^{-Z}.

To quantify the contribution of hidden neurons to the prediction discrepancy induced by the latent representation $\widehat{Z}$, we follow the idea of Integrated Gradients (IG) and use the zero vector as the baseline for hidden activations. Specifically, for the two input conditions with and without the latent representation, we construct continuous interpolation paths from the zero baseline to the actual hidden activations:

$$\widetilde{h}_{i,t}^{s}(\alpha) = \alpha h_{i,t}^{s}, \quad \alpha \in [0,1], \quad s \in \{+Z, -Z\}, \tag{6}$$

where $s \in \{+Z, -Z\}$ denotes the input conditions with and without the latent representation $\widehat{Z}$, respectively, and α is the interpolation coefficient. When $\alpha = 0$, the hidden activations are at the zero baseline; when $\alpha = 1$, they recover the actual hidden activations produced by the model.

Along the above integration paths, we recompute the model's predicted probability of the target output $Y = Yes$ under the two input conditions:

$$P_i^{+Z}(\alpha) = P\left(Y = Yes \mid \widetilde{H}_i^{+Z}(\alpha)\right), \quad P_i^{-Z}(\alpha) = P\left(Y = Yes \mid \widetilde{H}_i^{-Z}(\alpha)\right). \tag{7}$$

Accordingly, we define the prediction discrepancy along the integration path as:

$$D_i(\alpha) = P_i^{+Z}(\alpha) - P_i^{-Z}(\alpha), \tag{8}$$

where $D_i(\alpha)$ measures the prediction discrepancy between the conditions with and without the latent representation $\widehat{Z}$ as the hidden activations gradually recover from the zero baseline to their observed values.

For the input containing $\widehat{Z}_i$, the attribution score of the j-th neuron in the l-th layer is defined as:

$$BIG_i^{+Z}(j) = \frac{1}{T_i} \sum_{t=1}^{T_i} H_{i,l,t,j}^{+Z} \int_0^1 \frac{\partial D_i(\alpha)}{\partial H_{i,l,t,j}^{+Z}} d\alpha, \tag{9}$$

where T_i denotes the length of the input embedding sequence for the i-th sample, i.e., the total length of the soft prompt and the original text tokens.

Similarly, for the input without $\widehat{Z}_i$, the attribution score of the j-th neuron in the l-th layer is:

$$BIG_i^{-Z}(j) = \frac{1}{T_i} \sum_{t=1}^{T_i} H_{i,l,t,j}^{-Z} \int_0^1 \frac{\partial D_i(\alpha)}{\partial H_{i,l,t,j}^{-Z}} d\alpha. \tag{10}$$

Finally, we average the attribution results under the two conditions to obtain the paired attribution score of the j-th neuron for the i-th sample:

$$BIG_i(j) = \frac{BIG_i^{+Z}(j) + BIG_i^{-Z}(j)}{2}. \tag{11}$$

where $BIG_i(j)$ denotes the contribution of the j-th hidden neuron to the prediction bias induced by the latent confounders in the i-th sample.

In practice, the path integrals in Equations (9) and (10) are approximated using n-step Riemann summation. Specifically, the attribution scores under the with $\widehat{Z}$ and without $\widehat{Z}$ conditions are computed as follows:

$$BIG_i^{+Z}(j) \approx \frac{1}{T} \sum_{t=1}^{T} H_{i,l,t,j}^{+Z} \cdot \frac{1}{n} \sum_{k=1}^{n} \frac{\partial D_i(\alpha_k)}{\partial H_{i,l,t,j}^{+Z}}, \tag{12}$$

$$BIG_i^{-Z}(j) \approx \frac{1}{T} \sum_{t=1}^{T} H_{i,l,t,j}^{-Z} \cdot \frac{1}{n} \sum_{k=1}^{n} \frac{\partial D_i(\alpha_k)}{\partial H_{i,l,t,j}^{-Z}}, \tag{13}$$

where $\alpha_k = \frac{k}{n}$ denotes the k-th interpolation point, and n is the number of integration steps.

Given M paired samples, we aggregate the paired attribution scores across all samples to estimate the global importance of each neuron:

$$A_j = \frac{1}{M} \sum_{i=1}^{M} \left| BIG_i(j) \right|, \tag{14}$$

where A_j denotes the average contribution of the j-th neuron to the prediction discrepancy induced by the latent representation $\widehat{Z}$.

To rank neurons according to the magnitude of their contributions, we define

$$Score_j = A_j. \tag{15}$$

Finally, we select the top K neurons with the highest attribution scores:

$$\mathcal{N}_Z = TopK\left(\{Score_j\}\right), \tag{16}$$

where $\mathcal{N}_Z$ denotes the set of neurons identified as being associated with latent confounding bias.

3.4 Neuron Intervention and Causal Effect Estimation

After identifying the set of latent confounding bias-related neurons $\mathcal{N}_Z$, we perform causal intervention on their activations during inference. We then quantify the effectiveness of the intervention by estimating the average treatment effect (ATE) of the latent representation on the model output within an SCM.

3.4.1 ESTIMATING THE ATE BEFORE NEURON INTERVENTION

According to the causal graph in Fig. 1 (a), we first estimate the ATE of $\widehat{Z}$ on the model output. Specifically, $\mathbf{E}^{+Z}$ and $\mathbf{E}^{-Z}$ represent the two potential treatment states, corresponding to the inputs with and without the latent representation, respectively. The ATE is defined as:

$$ATE = \frac{1}{N}\sum_{i=1}^{N}\left[P(Y_i \mid do(\mathbf{E}_i = \mathbf{E}_i^{+Z})) - P(Y_i \mid do(\mathbf{E}_i = \mathbf{E}_i^{-Z}))\right]. \tag{17}$$

When $ATE \neq 0$, it indicates that, on average, the model output is affected by the latent representation encoded in the input.

3.4.2 ESTIMATING THE ATE AFTER NEURON INTERVENTION

For the identified bias-related neurons $\mathcal{N}_Z$, let H_Z denote their factual activations under the current input before intervention. We define a family of parameterized interventions as

$$do\big(H_Z = \hat{H}_Z\big), \qquad \hat{H}_Z = \gamma H_Z, \quad 0 \leq \gamma \leq 1,$$

where $\hat{H}_Z$ specifies the value imposed on the bias-related neurons under the intervention, and γ controls the intervention strength.

The post-intervention ATE is then defined as:

$$ATE_{\text{int}} = \frac{1}{N}\sum_{i=1}^{N}\left[P\left(Y_i \mid do(\mathbf{E}_i = \mathbf{E}_i^{+Z}, H_Z = \hat{H}_Z)\right) - P\left(Y_i \mid do(\mathbf{E}_i = \mathbf{E}_i^{-Z}, H_Z = \hat{H}_Z)\right)\right]. \tag{18}$$

To quantify the extent to which neuron intervention mitigates the influence of the latent representation, we define the change in the absolute ATE as:

$$\Delta ATE = |ATE| - |ATE_{\text{int}}|. \tag{19}$$

When $\Delta ATE > 0$, it indicates that the neuron intervention reduces the average causal effect of the latent representation on the model output.

3.5 BIAS NEURON SUPPRESSION AT INFERENCE TIME

During inference, given an input X, the LLM first computes its hidden representation $H = f_\theta(X)$. Let H_Z denote the factual activations of the identified bias-related neurons $\mathcal{N}_Z$. We then implement the intervention $(do\big(H_Z = \hat{H}_Z\big), \hat{H}_Z = \gamma H_Z, 0 \leq \gamma \leq 1)$ on $\mathcal{N}_Z$, while keeping the activations of all other neurons unchanged. In practice, this intervention is implemented by replacing the factual activations H_Z with the prescribed intervention value $\hat{H}_Z$ during the forward pass. We denote the resulting post-intervention representation as $\widehat{H}$. Finally, the post-intervention representation $\widehat{H}$ is passed through the LM head to obtain the debiased prediction:

$$\widehat{Y} = \text{LMHead}\big(\widehat{H}\big).$$

4 EXPERIMENT

To evaluate the effectiveness of the proposed CILC framework, we conducted extensive experiments on multiple real-world datasets and investigated the following research questions: **RQ1:** Can CILC effectively improve LLM recommendation performance under direct prediction scenarios? **RQ2:** Can CILC complement existing causal reasoning methods by mitigating latent-information-related interference in LLM reasoning? **RQ3:** Does the performance improvement of CILC stem from precise interventions on latent confounding bias-related neurons, rather than randomly intervening on arbitrary neurons?

4.1 EXPERIMENTAL SETUP

Datasets. We conduct experiments on eight real-world datasets: movie (Harper & Konstan, 2015), book (Ziegler et al., 2005), ML-1M (Harper & Konstan, 2015), Amazon-Book (He & McAuley, 2016), GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), CommonsenseQA (Talmor et al., 2019) and StrategyQA (Geva et al., 2021). Detailed descriptions of these datasets are provided in Appendix D.1.

Baseline Methods. Under direct prediction scenarios, we compare CILC with the following baselines: GRU4Rec (Hidasi et al., 2016), Caser (Tang & Wang, 2018), SASRec (Kang & McAuley, 2018), DROS (Yang et al., 2023), GRU-BERT (Devlin et al., 2019), DROS-BERT (Devlin et al., 2019), MF (Koren et al., 2009), LightGCN (He et al., 2020), ICL (Dai et al., 2023), Prompt4NR-V (Zhang & Wang, 2023), TALLRec (Bao et al., 2023), CoLLM-MF (Zhang et al., 2025b), and DDP (Li et al., 2025). Following their original settings, TALLRec and CoLLM-MF use LLaMA-7B and Vicuna-7B as backbone models, respectively. We incorporate DDP into TALLRec and CoLLM-MF, denoted as DDP-TALLRec and DDP-CoLMF. Similarly, integrating CILC into the two LLM-based recommenders yields CILC-TALLRec and CILC-CoLMF.

Under reasoning scenarios, we compare CILC with representative reasoning baselines, including ICL (Brown et al., 2020), CoT (Wei et al., 2022), CoT-SC (Wang et al., 2023b), SoT (Ning et al., 2023), CP (Zhang et al., 2025a), and ACPS (Li et al., 2026). Following ACPS, we use LLaMA-3.1 8B as the backbone model. CILC-ACPS is constructed by applying CILC-based neuron intervention to the bias-related neurons identified during ACPS inference. Detailed descriptions of all baselines are provided in Appendix D.2.

Evaluation Metrics. We adopt AUC, UAUC (Liu et al., 2021), and Accuracy (Acc) as evaluation metrics. To ensure the reliability of the experimental results, all experiments are repeated with five different random seeds, and the average of the five experimental results is reported.

Implementation Details. We implement CILC using PyTorch and optimize the model parameters with the AdamW optimizer (Kingma & Ba, 2015). More implementation details are provided in Appendix D.3.

Table 1: The AUC performance comparison of different methods under few-shot scenarios. Bold is used to highlight the best results, while the runner-up results are underlined.

Dataset	movie			book		
Few-shot / Methods	16	64	256	16	64	256
GRU4Rec	0.4907	0.4987	0.5289	0.4895	0.4964	0.4986
Caser	0.4968	0.5106	0.5420	0.4984	0.4972	0.4957
SASRec	0.5043	0.5048	0.5225	0.4948	0.5006	0.5020
DROS	0.5076	0.5154	0.5407	0.4928	0.4913	0.4913
GRU-BERT	0.5085	0.5165	0.5344	0.5007	0.4964	0.4979
DROS-BERT	0.5021	0.5171	0.5394	0.5007	0.4898	0.5020
TALLRec	0.6724	0.6748	<u>0.7198</u>	0.5636	0.6039	0.6438
DDP-TALLRec	<u>0.6783</u>	<u>0.6786</u>	0.7189	<u>0.5706</u>	<u>0.6062</u>	<u>0.6459</u>
CILC-TALLRec	**0.6907**	**0.6901**	**0.7281**	**0.5833**	**0.6125**	**0.6531**

4.2 OVERALL PERFORMANCE UNDER DIRECT PREDICTION SCENARIOS (**RQ1**)

4.2.1 FEW-SHOT TRAINING SETTING

We evaluate CILC under few-shot settings to examine its effectiveness in data-scarce scenarios. As shown in Table 1, CILC-TALLRec achieves the best performance across all datasets and settings. These improvements demonstrate that reducing latent confounding effects can enhance the prediction capability of LLM-based recommenders.

Table 2: The performance comparison of different methods under the full-sample setting. The best results are highlighted in bold, and the runner-up results are underlined.

Dataset	ML-1M		Amazon-Book	
Methods	AUC	UAUC	AUC	UAUC
MF	0.6482	0.6361	0.7134	0.5565
LightGCN	0.5959	0.6499	0.7103	0.5639
SASRec	0.7078	0.6884	0.6887	0.5714
ICL	0.5320	0.5268	0.4820	0.4856
Prompt4NR-V	0.7071	0.6739	0.7224	0.5881
CoLLM-MF	0.7295	0.6875	0.8109	0.6225
DDP-CoLMF	<u>0.7311</u>	<u>0.6881</u>	<u>0.8124</u>	<u>0.6237</u>
CILC-CoLMF	**0.7397**	**0.6972**	**0.8209**	**0.6313**

Table 3: Performance comparison of different methods on four datasets under reasoning scenarios. Acc is reported. The best results are shown in bold, while the runner-up results are underlined.

Method	GSM8K Acc	MATH Acc	ComQA Acc	StrQA Acc
ICL	0.1876	0.1875	0.2845	0.5691
CoT	0.3810	0.4035	0.6173	0.5280
CoT-SC	0.3077	0.4208	0.5714	0.6429
SoT	0.3778	0.4083	0.6076	0.5204
CP	0.6967	0.4667	0.6010	0.7210
ACPS	<u>0.7971</u>	<u>0.4680</u>	<u>0.6364</u>	<u>0.7303</u>
CILC-ACPS	**0.8103**	**0.4797**	**0.6495**	**0.7423**

4.2.2 Full-Data Training Setting

To further evaluate CILC under sufficient data settings, we conduct full-data experiments on the ML-1M and Amazon-Book datasets. As shown in Table 2, CILC-CoLMF consistently outperforms other methods on both datasets. This suggests that latent confounding information encoded in LLM parameters may still affect recommendation performance even with sufficient data, and CILC can effectively mitigate its impact.

4.3 Overall Performance under Reasoning Scenarios (**RQ2**)

To evaluate the effectiveness of CILC in reasoning scenarios, we conduct experiments on four datasets, with results summarized in Table 3. CILC-ACPS achieves the best performance across all datasets, suggesting that neuron-level intervention can effectively reduce latent-information-related interference and complement front-door adjustment in reasoning tasks.

In addition, we also conducted an ablation study (**RQ3**), causal effect analysis, hyperparameter sensitivity analysis, external validation using popularity-related metrics, and inference efficiency comparison. Detailed results are provided in Appendix E.1, E.2, E.3, E.4, and E.5, respectively.

5 Conclusion

In this work, we propose CILC, a causal intervention framework for latent confounding bias in LLMs. Specifically, CILC first uses a VAE to learn latent representations from user history information and maps them into soft prompts that can be injected into the LLM input space. Then, CILC employs the BIG to identify bias-related neurons by comparing differences in neuron activations under input conditions with and without the latent soft prompts. Finally, during LLM inference, targeted interventions are performed on these bias-related neurons, thereby effectively weakening the impact of latent confounding bias on LLM outputs. Extensive experimental results on three different LLM backbones and eight real-world datasets show that CILC can effectively mitigate the impact of latent confounding bias and improve predictive performance.

AI USE STATEMENT

In this work, we used generative AI tools to assist with improving the readability and clarity of the manuscript, including language polishing and grammatical refinement. We also used generative AI tools for assisting with debugging of experimental code. The proposed methodology, theoretical formulation, experimental design, implementation, and analysis were developed and conducted by the authors. All AI-assisted suggestions were carefully reviewed and verified by the authors. We take full responsibility for the final content of this work, including all claims, methodologies, experimental results, and conclusions.

REFERENCES

Keqin Bao, Jizhi Zhang, et al. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 1007–1014, 2023.

Tom Brown, Benjamin Mann, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, pp. 1877–1901, 2020.

Debo Cheng, Jiuyong Li, et al. Local search for efficient causal effect estimation. *IEEE Transactions on Knowledge and Data Engineering*, 35(9):8823–8837, 2022.

Debo Cheng, Jiuyong Li, et al. Data-driven causal effect estimation based on graphical causal modelling: A survey. *ACM Computing Surveys*, 56(5):1–37, 2024.

Debo Cheng, Jianfeng Deng, et al. Beyond factual queries: A novel predictive retrieval-augmented generation. In *Proceedings of the ACM Web Conference*, pp. 3870–3880, 2026.

Karl Cobbe, Vineet Kosaraju, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Sunhao Dai, Ninglu Shao, et al. Uncovering chatgpt's capabilities in recommender systems. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 1126–1132, 2023.

Jacob Devlin, Ming-Wei Chang, et al. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 4171–4186, 2019.

Yue Feng, Shuchang Liu, et al. A large language model enhanced conversational recommender system. *arXiv preprint arXiv:2308.06212*, 2023.

Yunfan Gao, Tao Sheng, et al. Chat-rec: Towards interactive and explainable llms-augmented recommender system. *arXiv preprint arXiv:2303.14524*, 2023.

Mor Geva, Daniel Khashabi, et al. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9: 346–361, 2021.

Roberto Gozalo-Brizuela and Eduardo C Garrido-Merchan. Chatgpt is not all you need. a state of the art review of large generative ai models. *arXiv preprint arXiv:2301.04655*, 2023.

Yue Guo, Yi Yang, et al. Auto-debias: Debiasing masked language models with automated biased prompts. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pp. 1012–1023, 2022.

F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, 5(4):1–19, 2015.

Ruining He and Julian McAuley. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web*, pp. 507–517, 2016.

Xiangnan He, Kuan Deng, et al. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 639–648, 2020.

Dan Hendrycks, Collin Burns, et al. Measuring mathematical problem solving with the math dataset. In *Thirty-fifth Conference on Neural Information Processing Systems*, 2021.

Balázs Hidasi, Alexandros Karatzoglou, et al. Session-based recommendations with recurrent neural networks. In *International Conference on Learning Representations*, pp. 1–10, 2016.

Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *IEEE International Conference on Data Mining*, pp. 197–206, 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, pp. 1–15, 2015.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Yehuda Koren, Robert Bell, et al. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

Manabu Kuroki and Judea Pearl. Measurement bias and effect restoration in causal inference. *Biometrika*, 101(2):423–437, 2014.

Bowen Li, Ziqi Xu, et al. Debiasing large language models via adaptive causal prompting with sketch-of-thought. In *Findings of the Association for Computational Linguistics*, pp. 4481–4499, 2026.

Jingling Li, Zeyu Tang, et al. Prompting fairness: Integrating causality to debias large language models. In *International Conference on Learning Representations*, pp. 63606–63629, 2025.

Xinhang Li, Chong Chen, et al. E4srec: An elegant effective efficient extensible solution of large language models for sequential recommendation. *arXiv preprint arXiv:2312.02443*, 2023.

Dawen Liang, Rahul G Krishnan, et al. Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 World Wide Web Conference*, pp. 689–698, 2018.

Paul Pu Liang, Chiyu Wu, et al. Towards understanding and mitigating social biases in language models. In *International Conference on Machine Learning*, pp. 6565–6576, 2021.

Jiayi Liao, Sihang Li, et al. Llara: Large language-recommendation assistant. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1785–1795, 2024.

Yujie Lin, Kunquan Li, et al. Bi-directional bias attribution: Debiasing large language models without modifying prompts. *arXiv preprint arXiv:2602.04398*, 2026.

Junling Liu, Chao Liu, et al. Is chatgpt a good recommender? a preliminary study. *arXiv preprint arXiv:2304.10149*, 2023.

Yiyu Liu, Qian Liu, Yu Tian, Changping Wang, Yanan Niu, Yang Song, and Chenliang Li. Concept-aware denoising graph neural network for micro-video recommendation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 1099–1108, 2021.

Jianxin Ma, Chang Zhou, et al. Learning disentangled representations for recommendation. In *Advances in Neural Information Processing Systems*, pp. 5711–5722, 2019.

Kevin Meng, David Bau, et al. Locating and editing factual associations in gpt. In *Advances in Neural Information Processing Systems*, pp. 17359–17372, 2022.

Kevin Meng, Arnab Sen Sharma, et al. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*, pp. 8694–8714, 2023.

Wang Miao, Zhi Geng, et al. Identifying causal effects with proxy variables of an unmeasured confounder. *Biometrika*, 105(4):987–993, 2018.

Moin Nadeem, Anna Bethke, et al. Stereoset: Measuring stereotypical bias in pretrained language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 5356–5371, 2021.

Roberto Navigli, Simone Conia, et al. Biases in large language models: origins, inventory, and discussion. *ACM Journal of Data and Information Quality*, 15(2):1–21, 2023.

Xuefei Ning, Zinan Lin, et al. Skeleton-of-thought: Large language models can do parallel decoding. In *Third Workshop on Efficient Natural Language and Speech Processing*, 2023.

Daisuke Oba, Masahiro Kaneko, et al. In-contextual gender bias suppression for large language models. In *Findings of the Association for Computational Linguistics*, pp. 1722–1742, 2024.

Long Ouyang, Jeffrey Wu, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, pp. 27730–27744, 2022.

Alicia Parrish, Angelica Chen, et al. Bbq: A hand-built bias benchmark for question answering. In *Findings of the Association for Computational Linguistics*, pp. 2086–2105, 2022.

Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2 edition, 2009.

Scott Sanner, Krisztian Balog, et al. Large language models are competitive near cold-start recommenders for language-and item-based preferences. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 890–896, 2023.

Timo Schick, Sahana Udupa, et al. Self-diagnosis and self-debiasing: A proposal for reducing corpus-based bias in nlp. *Transactions of the Association for Computational Linguistics*, 9:1408–1424, 2021.

Mukund Sundararajan, Ankur Taly, et al. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pp. 3319–3328, 2017.

Alon Talmor, Jonathan Herzig, et al. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 4149–4158, 2019.

Alex Tamkin, Amanda Askell, et al. Evaluating and mitigating discrimination in language model decisions. *arXiv preprint arXiv:2312.03689*, 2023.

Jiaxi Tang and Ke Wang. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pp. 565–573, 2018.

Romal Thoppilan, Daniel De Freitas, et al. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*, 2022.

Hugo Touvron, Thibaut Lavril, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Jesse Vig, Sebastian Gehrmann, et al. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, pp. 12388–12401, 2020.

Fei Wang, Wenjie Mo, et al. A causal view of entity bias in (large) language models. In *Findings of the Association for Computational Linguistics*, pp. 15173–15184, 2023a.

Xuezhi Wang, Jason Wei, et al. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, pp. 12648–12671, 2023b.

Jason Wei, Xuezhi Wang, et al. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, pp. 24824–24837, 2022.

Zhengyi Yang, Xiangnan He, et al. A generic learning framework for sequential recommendation with distribution shifts. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 331–340, 2023.

Shunyu Yao, Jeffrey Zhao, et al. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, pp. 30084–30116, 2023.

Congzhi Zhang, Linhai Zhang, et al. Causal prompting: Debiasing large language model prompting based on front-door adjustment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 25842–25850, 2025a.

Jizhi Zhang, Keqin Bao, et al. Is chatgpt fair for recommendation? evaluating fairness in large language model recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 993–999, 2023.

Yang Zhang, Fuli Feng, et al. Collm: Integrating collaborative embeddings into large language models for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 37(5): 2329–2340, 2025b.

Zizhuo Zhang and Bang Wang. Prompt learning for news recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 227–237, 2023.

Bowen Zheng, Yupeng Hou, et al. Adapting large language models by integrating collaborative semantics for recommendation. In *IEEE 40th International Conference on Data Engineering*, pp. 1435–1448, 2024.

Fan Zhou, Yuzhou Mao, et al. Causal-debias: Unifying debiasing in pretrained language models and fine-tuning via causal invariant learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pp. 4227–4241, 2023.

Cai-Nicolas Ziegler, Sean M McNee, et al. Improving recommendation lists through topic diversification. In *Proceedings of the 14th International Conference on World Wide Web*, pp. 22–32, 2005.

TABLE OF CONTENTS: APPENDIX

A RELATED WORKS **14**

A.1 Debiasing LLMs 14

A.2 LLM-Based Recommender Systems 14

B Assumptions for CILC **15**

C METHOD DETAILS **16**

C.1 Learning Latent Confounders 16

D EXPERIMENTAL DETAILS **17**

D.1 Dataset Details. 17

D.2 Baseline Methods. 18

D.3 Implementation Details. 19

E ADDITIONAL EXPERIMENTAL RESULTS **19**

E.1 Ablation Study (**RQ3**) 19

E.2 Causal Effect Analysis 20

E.3 Hyperparameter Analysis. 21

E.4 External Validation Using Popularity-Related Metrics 21

E.5 Inference Efficiency Comparison. 22

A RELATED WORKS

We mainly review the related literature from two perspectives: (1) Debiasing LLM; and (2) LLM-based recommender systems.

A.1 DEBIASING LLMS

Existing LLM debiasing methods can be roughly divided into two categories: one aims to mitigate bias caused by *observable confounders*, while the other focuses on bias caused by *latent confounders*.

(1) Mitigating bias caused by observable confounders.

Recent studies have extensively explored biases caused by observable confounders such as gender and race (Schick et al., 2021; Tamkin et al., 2023; Vig et al., 2020; Zhou et al., 2023; Wang et al., 2023a). For example, Prompting Fairness (Li et al., 2025) proposes a causal selection mechanism to regulate the propagation of causal paths of social information; ICGBS (Oba et al., 2024) uses constructed counterfactual statements to suppress gender bias; Auto-Debias (Guo et al., 2022) reduces disparities in demographic attribute prediction through a distribution alignment loss. However, these methods mainly focus on observable confounders and largely overlook bias caused by latent confounders.

(2) Mitigating bias caused by unobservable confounders.

To mitigate bias caused by latent confounders, recent studies introduce CoT or SoT reasoning as a mediator and use the front-door adjustment criterion to reduce bias. For example, CP (Zhang et al., 2025a) treats CoT as a mediator and estimates the unbiased causal effect through front-door adjustment; ACPS (Li et al., 2026) leverages SoT to improve the inference efficiency of front-door adjustment debiasing. Since the front-door adjustment criterion requires that latent confounders do not affect the generation of the reasoning process (e.g., CoT), these methods implicitly assume that latent confounders will not affect the generation of the reasoning process. However, latent confounders have already been encoded into model parameters during pretraining. Since CoT is generated by these model parameters, latent confounding bias may propagate into the reasoning process, making the front-door assumption difficult to satisfy. In contrast, the proposed CILC framework directly intervenes on the activations of bias-related neurons. In reasoning scenarios, it can alleviate the interference of latent confounders with the generation of the reasoning process (e.g., CoT), thereby better satisfying the conditions of the front-door adjustment criterion. In addition, CILC can also be used to mitigate latent confounding bias in direct prediction scenarios, further extending its scope of application. Unlike model editing methods such as ROME (Meng et al., 2022) and MEMIT (Meng et al., 2023), CILC performs activation-level interventions during inference to mitigate confounding bias, rather than modifying model parameters to edit knowledge.

A.2 LLM-BASED RECOMMENDER SYSTEMS

LLMs have attracted increasing attention in recommender systems (Dai et al., 2023; Liu et al., 2023; Feng et al., 2023; Liao et al., 2024; Cheng et al., 2026). For example, LLMRec (Sanner et al., 2023) integrates ID information into LLMs to address the cold-start recommendation problem; E4SRec (Li et al., 2023) combines traditional recommendation models with LLMs to further improve recommendation performance; Chat-Rec (Gao et al., 2023) incorporates user profiles and historical interaction information into LLM prompts to enable interactive recommendation; FaiR-

LLM (Zhang et al., 2023) uses a fine-tuned LLM to produce high-quality text embeddings and leverages these embeddings to improve ranking performance; LC-Rec (Zheng et al., 2024) jointly learns language semantics and collaborative signals during fine-tuning, thereby improving recommendation accuracy; CoLLM (Zhang et al., 2025b) further introduces collaborative information into the fine-tuning stage, improving performance in warm-start recommendation scenarios. However, existing LLM-based recommender systems mainly focus on incorporating additional user and item information or improving representation learning capabilities, while the potential impact of latent confounding encoded in LLM parameters remains largely unexplored.

In traditional recommender systems, VAE-based latent representation learning methods have been widely adopted to capture latent factors and alleviate bias issues (Liang et al., 2018; Ma et al., 2019). Inspired by this paradigm, CILC extends latent representation modelling to LLMs and combines it with neuron-level causal intervention to mitigate latent confounding-related information encoded in LLM parameters. The proposed CILC framework can be seamlessly integrated into existing LLM-based recommender systems, effectively reducing the impact of latent confounding bias while improving recommendation accuracy.

B Assumptions for CILC

To formally characterize the limitations of standard front-door adjustment, we present the following lemma.

Lemma 1 (Failure of Front-door Adjustment)**.** *In the SCM of Fig. 1(b), if the latent confounder Z affects the confounder-related neuron activations H_Z, and the reasoning chain R is influenced by H_Z, i.e., $Z \to H_Z \to R$, then R cannot serve as a valid front-door mediator. In this case, there exists an unblocked backdoor path $X \leftarrow Z \to H_Z \to R$, such that $R \not\perp_d Z$. Therefore, the front-door criterion is violated, and front-door adjustment yields biased estimates.*

Proof. Since $Z \to H_Z \to R$, the latent confounder Z directly influences the reasoning chain R through the H_Z. Therefore, the path $X \leftarrow Z \to H_Z \to R$ is an unblocked backdoor path from X to R. This violates the second condition of the front-door criterion. As a result, the effect of X on R is confounded by Z, and the standard front-door adjustment based on R is biased. □

Lemma 1 shows that front-door adjustment is invalid when Z affects the reasoning process through H_Z. In contrast, Proposition 1 shows that directly intervening on H_Z can reduce the backdoor effect without relying on the front-door assumption. To formalize the causal justification of CILC, we introduce the following assumptions. Let $\widehat{Z}$ be the latent-confounder proxy inferred from observable history, and let H_Z denote the proxy-associated neuron activations.

Assumption 1 (Proxy validity)**.** *The latent proxy representation $\widehat{Z}$, inferred from observable input X, captures the information in the latent confounder Z that is relevant to the activation of H_Z. Specifically, $\widehat{Z}$ preserves the Z-related information influencing H_Z, so that it can be used to localize the activation patterns associated with latent confounding. This assumption is inspired by the proxy variable literature, where observed proxies are assumed to retain sufficient information about unmeasured confounders for causal analysis (Kuroki & Pearl, 2014; Miao et al., 2018).*

Assumption 2 (Mediation through bias-related neurons)**.** *The effect of the latent confounder Z on the prediction Y, conditional on the task input X, is primarily mediated through the identified neuron activations H_Z. Thus, H_Z captures the dominant pathway $Z \to H_Z \to Y$. Other pathways through which Z may affect Y are assumed to be negligible relative to this pathway.*

Assumption 3 (Local intervention)**.** *The intervention on H_Z primarily targets the pathway associated with the identified latent-confounder contribution. The effects on other pathways are considered secondary within the local approximation.*

Assumption 4 (Local linear response)**.** *Within a local neighbourhood of the factual activation H_Z, the first-order variation of the model output with respect to the identified neuron activations can be approximated linearly. Higher-order effects are treated as residual components beyond the local approximation.*

Proof of Proposition 1. By Assumption 2, the dominant contribution of the latent confounder Z to the prediction Y is mediated through the identified neuron activations H_Z. By Assumption 3,

intervening on H_Z primarily modifies this latent-confounder-related pathway, while the effects on other pathways are treated as secondary within the local approximation. Let

$$f(H_Z) = \mathbb{E}[Y \mid H_Z, X]$$

denote the local response of the model output to the identified neuron activations. By Assumption 4, $f(H_Z)$ admits a first-order approximation around the factual activation H_Z. Therefore, for a local perturbation of H_Z,

$$f(\gamma H_Z) - f(H_Z) \approx (\gamma - 1)\nabla f(H_Z) H_Z,$$

where higher-order effects and contributions from non-targeted pathways are absorbed into the local approximation. The intervention

$$do(H_Z = \gamma H_Z)$$

scales the identified neuron activations by γ. Hence, the first-order contribution of the identified latent-confounder pathway is attenuated by the same factor. When the measured effect is dominated by this pathway, the post-intervention effect admits the local approximation

$$ATE_{\text{int}} \approx \gamma ATE.$$

Taking the magnitude gives

$$|ATE_{\text{int}}| \approx \gamma |ATE|,$$

and consequently,

$$\Delta ATE = |ATE| - |ATE_{\text{int}}| \approx (1 - \gamma)|ATE|.$$

Thus, reducing γ progressively attenuates the contribution of the identified latent-confounder pathway. □

C METHOD DETAILS

C.1 LEARNING LATENT CONFOUNDERS

According to the standard VAE (Kingma & Welling, 2013), we aim to learn the posterior distribution $p_\theta(\widehat{Z} \mid X)$ by introducing an approximate posterior distribution $q_\phi(\widehat{Z} \mid X)$ and optimizing it using variational inference.

The objective function is defined as the evidence lower bound (ELBO):

$$\mathcal{L}(\theta, \phi) = \underbrace{E_{q_\phi(\widehat{Z}|X)}\left[\log p_\theta(X \mid \widehat{Z})\right]}_{I} - \underbrace{KL\left(q_\phi(\widehat{Z} \mid X) \parallel p_\theta(\widehat{Z})\right)}_{II}, \tag{20}$$

where $X = [x_{u1}, x_{u2}, \ldots, x_{un}]$ denotes the historical interaction vector of user u, $\log p_\theta(X \mid \widehat{Z})$ is the log-likelihood of the generative model, and $KL(\cdot\|\cdot)$ denotes the Kullback-Leibler divergence, which measures the discrepancy between the approximate posterior and the prior.

The generative process can be decomposed as:

$$\log p_\theta(X, \widehat{Z}) = \log p_\theta(X \mid \widehat{Z}) + \log p_\theta(\widehat{Z}), \tag{21}$$

where $p_\theta(\widehat{Z})$ denotes the prior distribution of the latent confounders, typically set to a standard normal distribution $N(0, I)$. The term $p_\theta(X \mid \widehat{Z})$ represents the conditional distribution of the observed data given the latent confounders.

Furthermore, we make the following assumptions:

$$\begin{aligned} p_\theta(\widehat{Z}) &:= N(0, I), \\ q_\phi(\widehat{Z} \mid X) &:= N(\mu_\phi(X), \sigma_\phi^2(X)), \end{aligned} \tag{22}$$

where $\mu_\phi(X)$ and $\sigma^2_\phi(X)$ are the parameters predicted by the encoder network.

To compute term I in Eq. equation 20, we have:

$$\log p_\theta(X \mid \widehat{Z}) = \sum_{i=1}^{n} \Big[x_{ui} \log(\mu_z(\widehat{Z})_i) + (1 - x_{ui}) \log(1 - \mu_z(\widehat{Z})_i) \Big], \tag{23}$$

where $\mu_z(\widehat{Z})_i$ is the output of the decoder network, x_{ui} is a binary indicator denoting whether item is shown to user.

Term II in Eq. equation 20 is computed as follows:

$$KL\left(q_\phi(\widehat{Z} \mid X) \parallel p_\theta(\widehat{Z})\right) = \int q_\phi\left(\widehat{Z} \mid X\right) \log \frac{q_\phi\left(\widehat{Z} \mid X\right)}{p_\theta\left(\widehat{Z}\right)} d\widehat{Z} \tag{24}$$

By maximizing Eq. equation 20, we can derive an approximate posterior $q_\phi(\widehat{Z} \mid X)$ for recovering $\widehat{Z}$.

D EXPERIMENTAL DETAILS

D.1 DATASET DETAILS.

In the direct prediction scenarios, we follow the experimental setups of TALLREC and CoLLM. The construction and usage details of each dataset are described as follows:

movie. We follow the experimental setup of TALLREC (Bao et al., 2023) and use the MovieLens-100K dataset (Harper & Konstan, 2015). According to its preprocessing procedure, we take the most recent 10,000 interactions from the raw data and split them into training, validation, and test sets at a ratio of 8:1:1. For each user, the 10 interactions preceding the target item are used as the historical behavior sequence; consistent with TALLREC, interactions with ratings higher than 3 are regarded as positive "like" samples.

book. We adopt the BookCrossing preprocessing specification released by TALLREC (Ziegler et al., 2005). For each user, one item that the user has interacted with is randomly selected as the target item, and 10 items are randomly sampled from the remaining items to form the historical interaction sequence. The dataset is split into training, validation, and test sets at a ratio of 8:1:1. Ratings range from 1 to 10, and interactions with ratings greater than 5 are defined as positive "like" samples.

ML-1M. MovieLens-1M (Harper & Konstan, 2015) contains user ratings for movies ranging from 1 to 5. Following CoLLM's data processing strategy, we treat interactions with ratings greater than 3 as positive samples ("like") and organize the data in chronological order: records from the most recent 20 months are taken, with the first 10 months used for training, the next 5 months used for validation, and the last 5 months used for testing.

Amazon-Book. Amazon-Book (He & McAuley, 2016) is a book rating dataset commonly used in recommendation tasks, with ratings ranging from 1 to 5. We strictly adopt CoLLM's preprocessing settings and define interactions with ratings no less than 4 as positive samples ("like"). For the temporal split, we select data from the entire year of 2017, use the first 11 months as the training set, and evenly split the last month into validation and test sets.

In the reasoning scenarios, we follow the datasets and evaluation protocols of ACPS (Li et al., 2026), selecting four benchmarks: GSM8K, MATH, CommonsenseQA, and StrategyQA. ACPS itself inherits the experimental setup of CP (Zhang et al., 2025a), and the data splits and evaluation metrics for each dataset are kept consistent with it.

GSM8K. GSM8K (Cobbe et al., 2021) contains 8,500 grade-school math word problems, whose solutions require 2 to 8 arithmetic operations.

MATH. MATH (Hendrycks et al., 2021) covers 12,500 competition-level math problems, accompanied by detailed step-by-step solutions.

CommonsenseQA. CommonsenseQA (Talmor et al., 2019) contains 12,247 commonsense question-answering questions based on ConceptNet.

StrategyQA. StrategyQA (Geva et al., 2021) is a yes/no question-answering benchmark containing 2,780 questions that require implicit multi-step reasoning, and each question requires integrating multiple facts for inference.

D.2 Baseline Methods.

To evaluate the effectiveness of the proposed CILC method, we compare it with the following baseline methods:

(1) **GRU4Rec** (Hidasi et al., 2016): GRU4Rec uses a recurrent neural network (RNN) to model complete user session sequences, thereby generating accurate recommendations.

(2) **Caser** (Tang & Wang, 2018): Caser leverages a convolutional neural network (CNN) to model users' recent interaction sequences, capturing both long-term and short-term preferences.

(3) **SASRec** (Kang & McAuley, 2018): SASRec dynamically models user behavior sequences and identifies the information most relevant to the current recommendation target.

(4) **DROS** (Yang et al., 2023): DROS incorporates a distribution adaptation mechanism into the empirical risk minimization (ERM) paradigm to improve model generalization and robustness.

(5) **GRU-BERT**: GRU-BERT introduces BERT (Devlin et al., 2019) into the GRU4Rec architecture, enabling joint modeling of item textual semantics and sequential behavior patterns.

(6) **DROS-BERT**: DROS-BERT deeply integrates BERT into DROS's distribution adaptation framework, further improving robustness and generalization performance.

(7) **MF** (Koren et al., 2009): Matrix factorization (MF) factorizes the rating matrix into low-dimensional latent representations of users and items, and models user-item interactions via their inner product.

(8) **LightGCN** (He et al., 2020): LightGCN uses simplified graph convolution operations to capture high-order user-item interaction relationships.

(9) **ICL** (Dai et al., 2023): ICL performs in-context learning through natural language prompts, aligning the LLM's recommendation outputs with task instructions.

(10) **Prompt4NR-V** (Zhang & Wang, 2023): Prompt4NR constructs multiple prompt templates to enhance the robustness and accuracy of recommendations; its Prompt4NR-V variant uses Vicuna-7B as the backbone LLM.

(11) **TALLRec** (Bao et al., 2023): TALLRec reframes the recommendation task as an instruction-following task and optimizes the LLM-based recommender system through instruction tuning. We introduce the proposed CILC framework on top of TALLRec to mitigate latent confounding bias and denote the resulting model as **CILC-TALLRec**. Following the original TALLRec setup, CILC-TALLRec uses LLaMA-7B as the backbone LLM.

(12) **CoLLM** (Zhang et al., 2025b): CoLLM improves recommendation performance by injecting collaborative signals into the LLM input, where the CoLLM-MF variant uses MF as the collaborative module. We further integrate CILC into CoLLM-MF to mitigate latent confounding bias and denote the resulting model as **CILC-CoLMF**. Following the original CoLLM configuration, CILC-CoLMF uses Vicuna-7B as the backbone LLM.

(13) **DDP** (Li et al., 2025): DDP proposes a prompt adjustment strategy based on a selection mechanism according to the causal paths through which social information affects LLM decisions. We further integrate DDP into TALLRec and CoLLM-MF, and denote the resulting models as **DDP-TALLRec** and **DDP-CoLMF**, respectively.

(14) **ICL** (Brown et al., 2020): In-Context Learning (ICL) performs in-context learning via input-output examples and does not explicitly provide reasoning steps.

(15) **CoT** (Wei et al., 2022): Chain-of-Thought (CoT) introduces intermediate reasoning steps to support logical inference.

(16) **CoT-SC** (Wang et al., 2023b): CoT with Self-Consistency (CoT-SC) samples multiple reasoning paths and determines the answer by majority voting.

(17) **SoT** (Ning et al., 2023): Sketch-of-Thought (SoT) generates concise reasoning sketches, reducing token consumption while preserving key logic.

(18) **CP** (Zhang et al., 2025a): Causal Prompting (CP) uses front-door adjustment to estimate the causal effect between the prompt and the answer.

(19) **ACPS** (Li et al., 2026): ACPS leverages SoT to improve the efficiency of front-door adjustment-based debiased reasoning. We obtain **CILC-ACPS** by intervening on the bias-related neurons identified by CILC during ACPS inference.

D.3 Implementation Details.

We implement CILC using PyTorch and optimize the LLM with the AdamW optimizer (Kingma & Ba, 2015). All experiments are conducted on an NVIDIA GeForce RTX 4090 GPU running Ubuntu 22.04. For CILC-TALLRec, we follow the original TALLRec implementation and adopt LLaMA-7B as the backbone model. For CILC-CoLMF, we follow the CoLLM implementation and adopt Vicuna-7B as the backbone model. For CILC-ACPS, we follow ACPS and adopt LLaMA-3.1 8B as the backbone. Specifically, X corresponds to questions from the reasoning dataset. We freeze the LLM and use it only as a text encoder to obtain the question embedding, which serves as the input to the VAE. The VAE then infers the latent representation $\widehat{Z}$, which is used for soft prompt construction and neuron localization. For each reasoning dataset, the VAE, soft prompt generator, BIG-based neuron localization process, and intervention strength γ are instantiated independently using only the training and validation splits. During ACPS inference, we only apply the resulting neuron-level intervention, while the SoT reasoning and front-door adjustment components are inherited from the original ACPS pipeline.

E Additional Experimental Results

Table 4: Ablation Study of the CILC Method on Two Real-World Datasets. The AUC is reported. The best results are shown in bold.

Dataset	movie			book		
Few-shot / Methods	16	64	256	16	64	256
w/o BIG	0.6724	0.6748	0.7198	0.5636	0.6039	0.6438
w Low-BIG	0.6486	0.6412	0.6979	0.5457	0.5889	0.6311
w Random	0.6532	0.6544	0.7033	0.5514	0.5926	0.6320
w BIG	**0.6907**	**0.6901**	**0.7281**	**0.5833**	**0.6125**	**0.6531**

E.1 Ablation Study (**RQ3**)

To verify whether the performance improvement of CILC arises from accurately identifying bias-related neurons rather than simply suppressing arbitrary neurons, we conduct ablation studies under both direct prediction and reasoning scenarios. We compare the following four neuron selection strategies:

(1) w/o BIG: Remove the bias-related neuron identification module and perform no neuron intervention.

(2) w Low-BIG: Select the same number of neurons with the lowest BIG scores for intervention.

(3) w Random: Randomly select the same number of neurons for intervention.

(4) w BIG: Intervene only on the identified bias-related neurons $\mathcal{N}_Z$.

Direct Prediction Scenarios. The experimental results are shown in Table 4. **w/o BIG** leads to an obvious performance drop, while **w Low-BIG** performs the worst. **w Random** is better than **w**

Low-BIG but still worse than **w/o BIG**. This confirms that the gain does not come from intervening on arbitrary neurons, but from targeting neurons with high BIG attribution scores. In contrast, **w BIG** achieves the best performance across all settings, further verifying that CILC alleviates latent confounding bias through targeted neuron intervention rather than indiscriminate neuron modification.

Table 5: Ablation study on neuron selection strategies under reasoning scenarios. Acc is reported. The best results are highlighted in bold.

Method	GSM8K Acc	MATH Acc	ComQA Acc	StrQA Acc
w/o BIG	0.7971	0.4680	0.6364	0.7303
w Low-BIG	0.7694	0.4398	0.6141	0.7015
w Random	0.7711	0.4435	0.6199	0.7154
w BIG	**0.8103**	**0.4797**	**0.6495**	**0.7423**

Reasoning Scenarios. To verify whether the improvement of CILC under reasoning scenarios also stems from accurately identifying latent confounding bias-related neurons, we conduct an additional ablation study on four reasoning datasets. The results are shown in Table 5. Removing the bias neuron identification module (**w/o BIG**) leads to an obvious performance drop, indicating that the improvement of CILC-ACPS depends on accurately locating neurons related to latent confounding bias. Moreover, **w Low-BIG** performs even worse than **w/o BIG** on most datasets. Although **w Random** is better than **w Low-BIG**, it still underperforms **w BIG**, confirming that the gain does not come from intervening on arbitrary neurons, but from targeting neurons with high BIG attribution scores. In contrast, **w BIG** achieves the best performance across all four datasets, further verifying that CILC can alleviate latent confounding bias in reasoning scenarios through targeted neuron intervention rather than indiscriminate neuron modification.

E.2 Causal Effect Analysis

Table 6: We show the AUC and ΔATE results under different values of K.

Dataset	K	AUC	ΔATE
movie	50	0.6795	0.0063
	100	0.6888	0.0072
	150	0.6907	0.0116
	200	0.6874	0.0097
book	50	0.5744	0.0069
	100	0.5816	0.0113
	150	0.5833	0.0135
	200	0.5801	0.0109

To evaluate the debiasing capability of CILC, we vary the number of intervened neurons K under the 16-shot setting on the movie and book datasets. For each K, we report both the recommendation performance (AUC) and the decrease in causal effect (ΔATE). Since ΔATE measures the reduction in the causal influence of latent confounders under causal intervention, it directly reflects the degree to which latent confounding bias is mitigated. The experimental results are shown in Table 6.

(1) As K increases, AUC first improves and then decreases. The parameters in the model encode both latent confounders and task-relevant knowledge. When K is small, only a portion of the latent confounders are suppressed, resulting in limited debiasing. As K increases, more latent confounders are intervened on, and performance improves further. However, when K is too large, the task-relevant knowledge encoded in the neurons is also affected, leading to performance degradation. Therefore, an appropriate intervention scale is crucial for balancing debiasing and performance.

(2) Across different values of K, ΔATE is always positive, indicating that intervening on bias-related neurons can effectively reduce the causal effect of latent confounders on model outputs.

Table 7: Effect of latent representation input on prediction consistency.

Dataset	P^{+Z}	P^{-Z}	$\lvert P^{+Z} - P^{-Z} \rvert$
movie (before)	0.6591	0.6398	0.0193
movie (after)	0.7095	0.7018	0.0077
book (before)	0.6813	0.6604	0.0209
book (after)	0.7166	0.7092	0.0074

To further illustrate the debiasing effect, Table 7 reports the average prediction probabilities under inputs with and without latent representations when $K = 150$. Since the causal effect is computed as the prediction difference between the two types of inputs, the reduction in the prediction gap $|P^{+Z} - P^{-Z}|$ intuitively explains the reduction in causal effect. After neuron intervention, the prediction gaps on both datasets are substantially reduced, indicating that CILC effectively mitigates the influence of latent confounders on outputs.

ΔATE arises from causal intervention on latent confounding bias and directly quantifies the reduction in causal influence caused by latent confounding bias. Therefore, a positive ΔATE indicates that CILC effectively mitigates latent confounding bias.

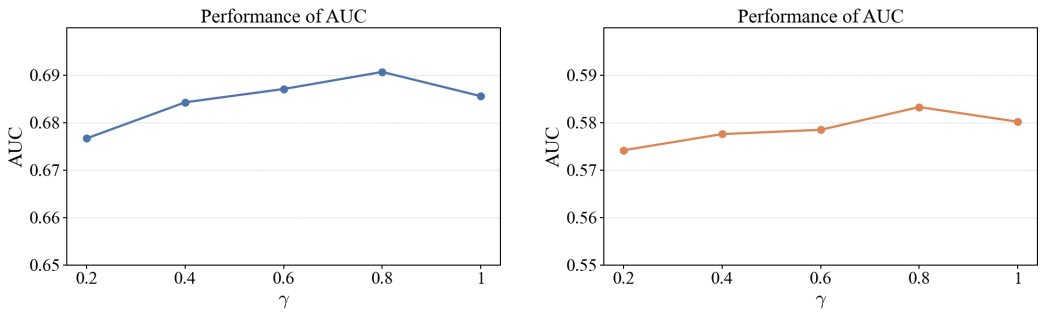
E.3 Hyperparameter Analysis.

To evaluate the impact of hyperparameters on the performance of the proposed CILC framework, we conduct a hyperparameter sensitivity analysis on the movie and book datasets under the 16-shot setting. We select the hyperparameters K on the validation set, and keep them fixed for all test evaluations. We vary only the hyperparameter γ within $[0, 1]$.

As shown in Figure 3, the experimental results show:

(1) movie dataset. As γ increases, AUC first improves and then decreases, achieving the best performance at $\gamma = 0.8$. This indicates that neurons encode both latent confounders and task-relevant knowledge. When γ is too small, latent confounders cannot be sufficiently intervened on. When γ is too large, task-relevant knowledge is affected. Therefore, a moderate value of γ provides a better trade-off between mitigating latent confounding bias and preserving task-relevant knowledge.

(2) book dataset. A similar trend is observed on the book dataset: AUC first rises and then falls, achieving the best performance at $\gamma = 0.8$. This result indicates that moderate suppression of bias-related neurons can also effectively mitigate latent confounding bias.



(a) (b)

Figure 3: Effect of the γ selection. (a) We show the results of AUC on the movie datasets. (b) We show the results of AUC on the book datasets.

E.4 External Validation Using Popularity-Related Metrics

To examine whether the effect of CILC goes beyond the self-inferred latent proxy $\widehat{Z}$, we conduct an external criterion validation using item popularity. Item popularity is computed independently

Table 8: External validation using popularity-related metrics. Lower is better for AvgPop@5 and PopBias@5, and higher is better for Coverage@5.

Method	AvgPop@5 ↓	PopBias@5 ↓	Coverage@5 ↑
TALLRec	201.20	85.37	0.1613
CILC-TALLRec	**188.74**	**74.74**	**0.1751**

Table 9: Runtime comparison between CILC-TALLRec and TALLRec on the movie dataset.

Method	Inference Time (minutes)
TALLRec	1.05
CILC-TALLRec	1.09

from the observed interaction data and is not used in constructing $\widehat{Z}$. Specifically, under the 16-shot setting on the movie dataset, we compare TALLRec and CILC-TALLRec using three popularity-related metrics (AvgPop@5, PopBias@5, and Coverage@5). For a fair comparison, both models are evaluated at the same positive prediction rate, and their Top-5 recommendation behaviors are then compared.

As shown in Table 8, CILC-TALLRec consistently shifts the recommendation behavior toward lower popularity. AvgPop@5 decreases from 201.20 to 188.74, indicating that after intervention, the average popularity of items receiving positive predictions in the Top-5 list is reduced. PopBias@5 decreases from 85.37 to 74.74, suggesting that the popularity level of recommended items is closer to that reflected by the user's historical interactions. Coverage@5 increases from 0.1613 to 0.1751, indicating that a wider range of items receives positive recommendations.

In summary, these results indicate that the effect of CILC is not limited to reducing the model's sensitivity to the self-inferred latent proxy $\widehat{Z}$.

E.5 Inference Efficiency Comparison.

Under the 16-shot setting on the movie dataset, we compare the inference efficiency of the original TALLRec and CILC-TALLRec. As shown in Table 9, the inference time of CILC-TALLRec does not increase substantially, indicating that the proposed CILC framework introduces only minimal inference overhead.