Machine Learning (2026) 115:59
https://doi.org/10.1007/s10994-026-07011-3

How Robust are Large Language Models Against Word-Level Spurious Correlations? A Causal Discovery Approach

Xin Miao[1] · Yongqi Li[1] · Hankun Kang[1] · Mayi Xu[1] · Jintao Wen[1] · Yuyang Ren[1] · Tieyun Qian[1,2]

Received: 14 July 2025 / Revised: 19 October 2025 / Accepted: 9 February 2026

Abstract
Large language models (LLMs) excel in information extraction (IE). However, word-level spurious correlations have been observed in prediction errors. Assessing LLMs' robustness against such risks is crucial for building reliable IE systems. Yet, spurious correlations within LLMs' semantics remain unevaluated, as existing studies detect them by relying on co-occurrence statistics from training data, which are unavailable for LLMs. To address this challenge, we propose a novel module **LLM Causal Discovery** (**LCD**), which leverages statistics encoded in model parameters to identify spurious correlations, grounded in causal discovery. Building on LCD, we introduce a framework **Spurious Correlation Evaluator** (**SCE**) to assess robustness using noisy data containing the identified spurious correlations. Our findings, evaluated on state-of-the-art (SOTA) LLMs, reveal their notable fragility and heavy reliance on statistical features. SCE's perturbation largely outperforms recent robustness evaluation strategies, establishing an SOTA attack system. Additionally, evaluation results can serve as effective feedback to enhance robustness. In summary, SCE tackles the challenge of quantifying word-level spurious correlations in LLMs, providing support for risk control. As a causality-based framework, it also evidences model stability and interpretability. Code and data are available at: https://github.com/NLPWM-WHU/SCE.

Keywords Natural language processing · Large language models · Spurious correlation · Robustness · Causal discovery · Information extraction

Editor: Mingming Gong.

✉ Tieyun Qian
qty@whu.edu.cn

Xin Miao
miaoxin@whu.edu.cn

Yongqi Li
liyongqi@whu.edu.cn

[1] School of Computer Science, Wuhan University, No. 299 Bayi Road, Wuhan 430072, China

[2] Intellectual Computing Laboratory for Cultural Heritage, Wuhan University, No. 299 Bayi Road, Wuhan 430072, China

Published online: 09 February 2026

1 Introduction

Correlation is not causation. — Karl Pearson (1857–1936)

Large language models (LLMs) excel at information extraction (IE) (Sarawagi et al., 2008), which involves extracting predefined information from unstructured text (Lu et al., 2022), including event detection (ED), entity typing (ET), and relation extraction (RE) (Huang et al., 2023; Wan et al., 2023; Wang et al., 2023). Owing to their vast knowledge (Chen, 2023; Zhao et al., 2023), LLMs are increasingly applied across diverse domains, including high-risk areas such as medicine and finance (Goel et al., 2023; Huang et al., 2023; Li et al., 2023; Monajatipoor et al., 2024; Rajpoot & Parikh, 2023), where robustness is a primary concern. However, researchers have observed an unexpected phenomenon (Zhang et al., 2024): entities are correlated with irrelevant labels that lack semantic grounding. As shown in the real case in Fig. 1c, such fragility is triggered by just a single word. This reflects **word-level spurious correlations** (Simon, 1954), formed by frequent word co-occurrence in training corpora (Elazar et al., 2022; Kang et al., 2023; Zhou et al., 2024). Such superficial statistical patterns are a key source of instability (Cui & Athey, 2022), threatening their reliability and safety (Raza et al., 2025; Yuan et al., 2025) in real-world scenarios (Geirhos et al., 2020).

Therefore, assessing the robustness of LLMs against such spurious correlations is crucial for developing reliable IE systems (Baliunas, 2023; Mondal & Sancheti, 2024). Specifically, a systematic evaluation offers both practical and academic value across three aspects. (1) **Risk Awareness**. It quantifies spurious correlations and supports the identification and management of latent risks (Gan et al., 2024; Sakib et al., 2024) for real-world applications. (2) **Trend Visibility**. It facilitates ongoing monitoring of model evolution and provides evidence to guide stability-oriented research (Liu et al., 2025; Wu et al., 2024). (3) **Interpretability**. It offers insights into models' learning mechanisms, improving their transparency and interpretability (Ashwani et al., 2024; Hobbhahn et al., 2022; Hu et al., 2024).

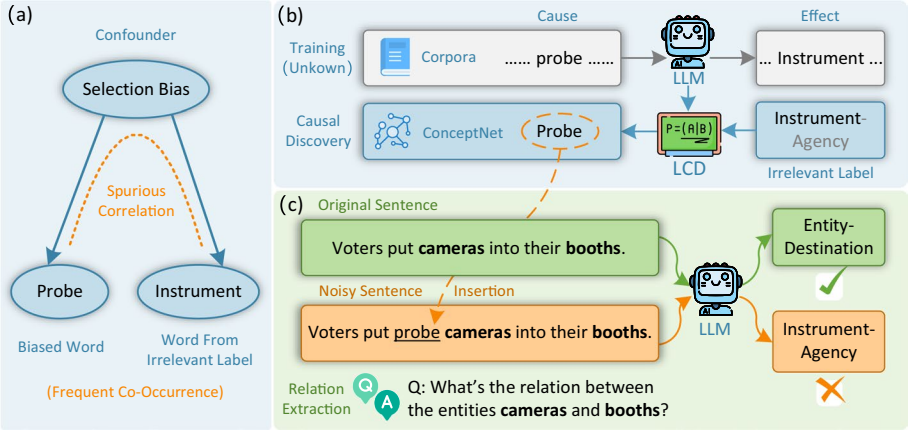

Fig. 1 An example of spurious correlation from RE. **a** The directed edges indicate the text generation process. **c** Bold denotes the given entities, and underlined indicates the inserted biased word

However, current LLM evaluations fall short in exploring this issue. Causal evaluations primarily focus on formal reasoning Jin et al. (2024a, 2024b), ignoring causality in semantics. Robustness evaluations rely on heuristic perturbations (Hu et al., 2025; Xiao et al., 2024; Zheng et al., 2024), overlooking spurious correlations. A causal lens (Pearl, 2000, 2009; Pearl & Mackenzie, 2018) on language modeling (Radford et al., 2019) is essential to address this issue, as spurious correlations stem from selection bias in training corpora (Ye et al., 2024). As shown in Fig. 1a, selection bias acts as a confounder (Pearl, 2009), promoting the co-occurrence of words *Probe* and *Instrument*. Furthermore, due to the autoregressive nature of LLMs, this co-occurrence becomes directional. As shown in Fig. 1b, *probe* frequently precedes *instrument*, causing the LLM to learn a spurious correlation where *probe* triggers *instrument* generation without considering context, as neural networks tend to learn shortcuts (Geirhos et al., 2020). In this example, *instrument* belongs to an irrelevant label, making *probe* a **biased word**: one that drives an LLM to generate an irrelevant label regardless of context. Our goal is to identify such biased words and insert them to create noisy data for robustness evaluation, as shown in Fig. 1c.

Identifying biased words is crucial to achieving this goal. Existing studies on spurious correlations in language models identify such words based on word co-occurrence statistics in training data (Elazar et al., 2022; Kang et al., 2023; Zhou et al., 2024). However, the training corpora of LLMs are inaccessible. From a generative perspective, biased words lead to the generation of irrelevant labels. We therefore frame this as a **causal discovery problem**: given an irrelevant label, identify words that cause the LLM to generate it without context. Causal discovery techniques (Spirtes & Glymour, 2000; Verma & Pearl, 1990) enable the identification of causal relationships from statistical independencies between variables. However, existing methods fall short because they estimate such independencies from observational data (Abdulaal et al., 2023; Liu et al., 2024; Long et al., 2023). To address this challenge, we treat LLMs as parameterized corpora for estimating statistical independence, as their parameters implicitly encode the statistics of training data (Carlini et al., 2021; Hammoudeh & Lowd, 2024). Specifically, disregarding context, we treat words as variables and use the LLM to estimate conditional probabilities among them. These probabilities are then used in conditional independence tests (Pearl, 2000, 2009) to infer their causal relationships in generation. Finally, words causing irrelevant labels are considered biased words. However, LLMs face two main difficulties when performing such tests: (1) as autoregressive models, they cannot estimate joint probabilities,[1] and (2) lack of a probability estimation method.

To address these difficulties, we propose two core techniques: (1) introducing conditional independence criteria tailored to the autoregressive nature of LLMs, and (2) developing an LLM-based probability estimation method that integrates prompts and model logits. These techniques enable LLMs to perform conditional independence tests effectively. Building on this, we propose a novel module, **LLM Causal Discovery** (**LCD**), which performs causal discovery using LLMs and identifies biased words for a given irrelevant label. As shown in Fig. 1b, given *Instrument* from the irrelevant label, LCD identifies a generative cause, *Probe*, from ConceptNet (Speer et al., 2017). Due to the absence of context, the generation is driven by a spurious correlation, and *Probe* is identified as a biased word. Building upon LCD, we introduce a robust evaluation framework, **Spurious Correlation Evaluator** (**SCE**). Specifically, SCE applies LCD to each correctly predicted instance to identify its biased words, and

[1] A proof by contradiction is presented in Appendix A.1.

inserts them individually into the original sentence to induce spurious correlations and create noisy data. To ensure label consistency and minimize interference, only one biased word is inserted at a time, following predefined task-specific rules. As shown in Fig. 1c, since RE relations are context-based, the biased word *probe* is inserted as an entity modifier to construct a noisy sentence, without altering the original relation between the given entities *cameras* and *booths*. Finally, we evaluate robustness using the noisy data.

Based on SCE, we conduct extensive experiments on state-of-the-art (SOTA) LLMs across three IE tasks: ED, ET, and RE (Ding et al., 2021; Grishman et al., 2005; Hendrickx et al., 2010). The results reveal three key findings that reflect the practical and academic value of SCE from three perspectives. (1) **Risk Awareness**. All evaluated LLMs are fragile to spurious correlations, with RE being particularly affected. On the widely used lightweight model Qwen-2-7B, the instability ratio reached 51.84%. While even the most advanced Qwen3-32B and GPT-4o show notable instability ratios of 30.34% and 18.18%, respectively. These results underscore the high risk of LLMs in IE applications, especially those involving multiple entities. (2) **Trend Visibility**. Increasing model parameters and training data size partially alleviate the issue, but current scaling laws face a bottleneck. For instance, Qwen-3, trained on a larger corpus, exhibits increased instability, indicating that scale alone is insufficient to address spurious correlations. Moreover, the SOTA performance of SCE over recent robustness evaluation methods suggests its potential as a long-term evaluation approach. For example, on Llama-3-8B, the perturbation effectiveness increased by at least 20.96% on average across three tasks. (3) **Interpretability**. SCE transparently reveals spurious correlations within LLMs' semantic knowledge, highlighting their continued reliance on statistical features (Wu et al., 2024). Furthermore, SCE helps diagnose model weaknesses and offers effective feedback for robustness enhancement. Notably, counterfactual examples (Miao et al., 2023, 2024; Zhang et al., 2023) generated using SCE feedback yield the most improvements. For instance, the stable accuracy of Llama-3-8B increases by 5.83%.

In summary, our key contributions are as follows:

- To fill the gap in evaluating semantic spurious correlations of LLMs, we introduce the **SCE** framework, which leverages statistics in model parameters to perform causal discovery and robustness evaluation without relying on training data, offering a causal lens for probing LLMs' stability and interpretability.
- Despite the close link between causality and stability, recent studies on LLM robustness largely lack a causal perspective, a gap that SCE aims to fill. SCE significantly outperforms recent evaluation methods, establishing an SOTA attack system.
- To identify spurious correlations, we propose the **LCD** module, which includes two novel techniques: (1) LLM-based conditional independence test and (2) LLM-based probability estimation. It is the first work introducing spurious correlation detection based on the LLM generation probability, laying a foundation for future research.
- We find that the evaluation results of SCE provide effective feedback for improving robustness. Notably, counterfactual examples generated under the guidance yield the most significant improvements, offering potential directions for future research.

The remainder of this paper is organized as follows. Section 2 briefly reviews related work. Section 3 introduces essential causal preliminaries. Section 4 presents the proposed SCE

framework and its core module, LCD. Section 5 reports extensive evaluation experiments and detailed discussions of the results. Section 6 provides further analytical studies and discussions. Finally, Sect. 7 concludes the paper.

2 Related Work

2.1 Word-Level Spurious Correlations

Word-level spurious correlations were often observed in pretrained models, such as hate speech detection (Tiwari et al., 2022), sentiment analysis (Wang et al., 2023), and RE (Wang et al., 2022; Yu et al., 2023). The transparency of fine-tuning makes such patterns easier to observe and to detect using co-occurrence statistics in training data (Elazar et al., 2022; Kang et al., 2023; Zhou et al., 2024). In the LLM era, however, the lack of transparency makes research difficult. While recent studies have reported spurious correlations in IE (Zhang et al., 2024), systematic evaluation remains lacking. We address this by probing word-level spurious correlations within LLMs without relying on training data.

2.2 Causal Evaluation for LLMs

Causal evaluation of LLMs can be approached in two ways, including empirical knowledge evaluation (Gao et al., 2023; Kiciman et al., 2024; Kim et al., 2023; Romanou et al., 2023; Willig et al., 2023) and causal reasoning evaluation (Jin et al., 2024a, b). The former posits that causal capability derives from empirical knowledge. Hence, they evaluate LLMs via cause-effect questions. However, LLMs may parrot causality rather than understand it Zecevic et al. (2023). In contrast, the latter synthesizes formalized questions based on classical causal structures (Pearl, 2009) to assess how LLMs grasp pure causal reasoning skills, i.e., the frontdoor adjustment and backdoor adjustment (Pearl & Mackenzie, 2018). However, they overlook the causality within LLMs' semantics, which is crucial for reliable semantic understanding (Wu et al., 2024). We investigate spurious correlations within LLMs' semantics, filling this gap.

2.3 Robustness Evaluation for LLMs

Current robustness evaluations for LLMs can be classified into two types: classical-based (Xiao et al., 2025; Wang & Zhao, 2024; Zhu et al., 2023) and LLM-specific (Hu et al., 2025; Xiao et al., 2024; Zheng et al., 2024). Classical-based methods primarily integrate traditional perturbation methods (Jin et al., 2020; Li et al., 2019; Naik et al., 2018; Ren et al., 2019; Ribeiro et al., 2020) to create hybrid noisy data, such as synonym replacement and character swapping. However, such commonly found errors pose very limited interference to LLMs (Singh et al., 2024), as web data may already expose them. Furthermore, LLM-specific methods introduce semantic distractions that are unfamiliar to LLMs based on heuristics, including neologisms, emojis, and semantically confusable expressions. Although they attempt to identify flaws in LLMs' knowledge, they lack a causal perspective and overlook the primary source of instability, i.e., spurious correlations (Cui & Athey, 2022). In this paper, we assess the robustness of LLMs through a causal lens.

2.4 LLMs for Causal Discovery

Due to their extensive knowledge, LLMs are primarily used as assistants for causal discovery. Initially, researchers explored knowledge-driven causal discovery using LLMs, where they construct complete causal graphs relying on LLMs (Jiralerspong et al., 2024; Long et al., 2022). However, these structures are unreliable (Wan et al., 2024), as their knowledge contains errors. Thus, researchers have explored data-driven methods recently, where the graph skeleton is identified by traditional methods, and edge directions are inferred by LLMs (Abdulaal et al., 2023; Liu et al., 2024; Long et al., 2023). They focus on utilizing LLMs for causal discovery, rather than applying causal theory to analyze the models themselves. We are the first to perform causal discovery within LLMs.

3 Preliminaries on Causality

3.1 Structural Causal Model

The structural causal model (SCM) (Pearl, 2000, 2009; Pearl & Mackenzie, 2018), i.e., causal graph, is often formalized as a directed acyclic graph (DAG) $\langle G := V, E \rangle$. Nodes V represent variables, while directed edges E denote cause-effect relationships. Parent nodes correspond to causes, and child nodes represent the resulting effects. For instance, consider the V-structure shown in Fig. 2a, $A \to C$ denotes that A has a causal effect on C. In this paper, a causal graph is employed to describe relationships in text generation. Nodes represent textual units, such as words or sentences. Directed edges indicate their generation relationship, either generated by a model or manually inserted. For instance, $A \to C$ indicates that the biased word A drives an LLM to generate the irrelevant label C. $A \to D$ denotes inserting A to generate the noisy sentence D, as shown in Fig. 2b.

3.2 V-structure and Causal Discovery

A V-structure describes a scenario where two variables simultaneously influence a common effect, known as a collider (Pearl, 2009), as shown in Fig. 2a. In this paper, we aim to discover the V-structure with an irrelevant label C as the collider, whose parent nodes A and B drive an LLM to generate the irrelevant label without context and thus are identified as biased words. Since V-structure uniquely defines a Markov equivalence class (Andersson et

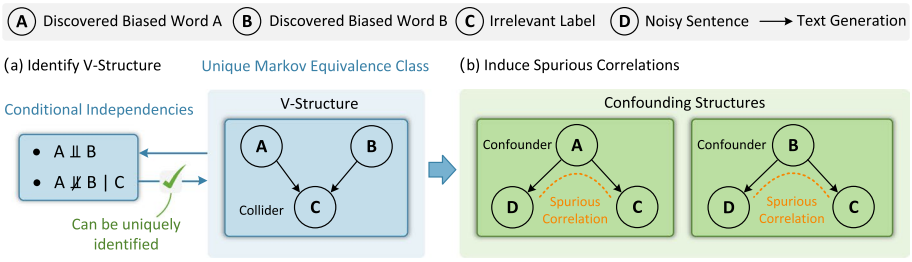

Fig. 2 Causal graphs of core ideas. **a** Identifying a V-structure where an irrelevant label acts as a collider. **b** Each discovered word is inserted into the noisy sentence, inducing a spurious correlation

al., 1997), it can be identified by the conditional independencies it implies. Thus, it is commonly used for causal discovery (Spirtes & Glymour, 2000; Verma & Pearl, 1990) via the conditional independence test:

$$P(A \mid B) = P(A) \Rightarrow A \perp\!\!\!\perp B, \tag{1}$$

$$P(A \mid B, C) \neq P(A \mid C) \Rightarrow A \not\perp\!\!\!\perp B \mid C, \tag{2}$$

where $\perp\!\!\!\perp$ and $\not\perp\!\!\!\perp$ denote statistical independence and statistical dependence, respectively. Equation 1 indicates that words A and B are statistically independent for an LLM, as given B does not affect the probability of it generating A. Equation 2 denotes that A and B are statistically dependent given C, as knowing B changes the probability of generating A in the condition of C. If these words satisfy such independencies for the LLM, a V-structure is identified, thereby discovering two biased words for it.

3.3 Confounding and Spurious Correlation

A confounding structure refers to a scenario where a variable influences two others, as shown in Fig. 2b. The common cause is known as a confounder (Pearl, 2009), which induces a spurious correlation between the affected variables. During noisy data construction, we insert the identified biased words A and B separately to create the noisy sentence D, which obtains a spurious correlation with the irrelevant label C. This is because the biased word A or B affects the generation of both D and C, serving as a confounder, which implicitly links them and introduces its spurious correlation with the irrelevant label into the noisy sentence. Finally, the constructed noisy data is used for evaluation.

3.4 Causal Assumptions

To enable causal discovery within LLMs, we adopt three commonly used assumptions in the LLM context (Wu et al., 2024; Zhang et al., 2025; Zhao et al., 2025). First, the modularity assumption (Pearl, 2009) posits that each causal mechanism in a system is autonomous. This means that each word, as a unit of semantic knowledge, can be considered and manipulated in isolation. Second, the faithfulness assumption (Pearl, 2009) holds that if two variables v_i and v_j are conditionally independent in the data, then all causal paths between them in the SCM are d-separated by some variable set. This enables the use of observed conditional independencies to infer causal structure. Third, the causal sufficiency assumption (Spirtes & Zhang, 2016) assumes that the variable set V includes all direct causes for each variable. That is, unobservable confounders such as data selection bias are no longer considered. This assumption simplifies causal discovery and makes it more feasible for black-box models, e.g. LLMs.

4 Methodology

In this section, we introduce the evaluation framework **Spurious Correlation Evaluator** (**SCE**). We first define the task, then present the overall framework, pipeline, and evaluation metrics, followed by a detailed description of the **LLM Causal Discovery** (**LCD**) module, which is the core component of the framework.

4.1 Task Formulation

4.1.1 IE Tasks

In this study, we select three commonly used IE tasks: event detection (ED) (Grishman et al., 2005), entity typing (ET) (Ding et al., 2021), and relation extraction (RE) (Hendrickx et al., 2010). Specifically, ED reflects fact-level semantics, ET captures category-level semantics, and relation extraction focuses on relation-level semantics. See Sect. 5.1 for more task explanation.

Let $\mathcal{X} = \{(x_i = (s_i, e_i), y_i)\}$ be a certain dataset, where $x_i \in \mathcal{X}$ denotes the *i*-th instance containing a sentence s_i and the given entity (or entities) e_i within it, and $y_i \in \mathcal{Y}$ denotes the corresponding label. Given an instance x_i, IE aims to predict its label y_i. For example, RE focuses on predicting the relation between given entities.

4.1.2 Robustness Evaluation

We define an incorrect prediction as $LLM : w_i \mapsto \hat{y}_i$, denoting that given a word w_i, the LLM are driven to generate an irrelevant label $\hat{y}_i$, where $\hat{y}_i \in \{\hat{y}_i \in \mathcal{Y} \mid \hat{y}_i \neq y_i\}$. To evaluate the robustness, we first identify such biased words $\mathcal{W}_i$ for all $\hat{y}_i$. Then we separately inserts each biased word $w_i \in \mathcal{W}_i$ into the original sentence s_i while ensure label consistency, to create noisy data $\mathcal{X}'_i = \{(x'_i = (s_i \oplus w_i, e_i), y_i) \mid w_i \in \mathcal{W}_i\}$ for x_i, where $\oplus$ indicates insertion perturbation. Finally, we assess the robustness using the noisy set $\mathcal{X}' = \{\mathcal{X}'_i \mid (x_i \in \mathcal{X}) \wedge (LLM : x_i \mapsto y_i)\}$, where $LLM : x_i \mapsto y_i$ involves the instances that the LLM predict correctly.

4.2 Overview of SCE

Figure 3a provides an overview of the SCE framework through a causal graph. Given an instance that the LLM predicts correctly, our objective is to introduce a spurious correlation between its input sentence and an irrelevant label. To this end, we first select an irrelevant label and employ the LCD module to identify a biased word through causal discovery, which drives the LLM to generate the irrelevant label. This process corresponds to the edge *Biased Word* → *Irrelevant Label*. Then, we insert the identified biased word into the original sentence to create a noisy one, corresponding to the edge *Biased Word* → *Noisy Sentence*. At this point, the biased word acts as a confounder that links the noisy sentence to the irrelevant label, thereby introducing the spurious correlation with the irrelevant label into the noisy sentence. Finally, the constructed noisy data is employed to evaluate the robustness of LLMs.

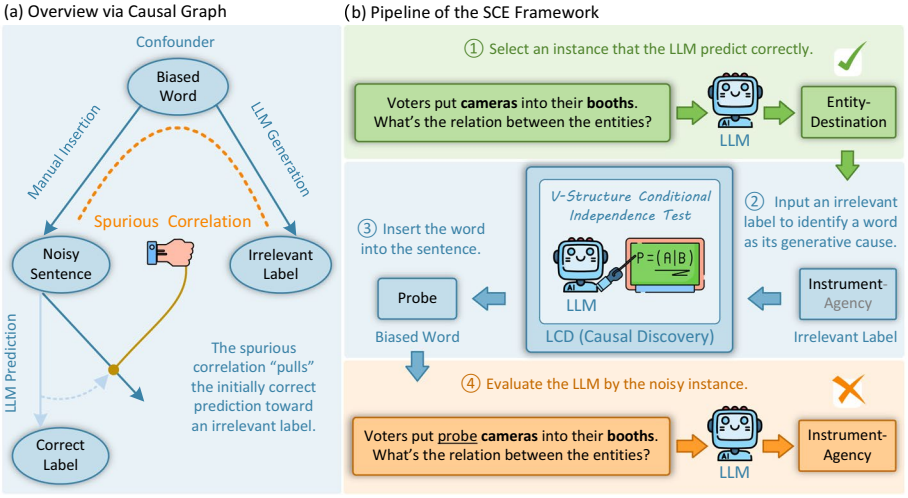

Fig. 3 Illustration of the SCE framework. **a** In the causal graph, blue arrows denote generation relationships, while the orange line indicates a spurious correlation. **b** In the pipeline, a RE example is provided. **Bold** indicates the given entities, and underlined denotes the inserted biased word

4.3 Pipeline of SCE

Figure 3b depicts the overall pipeline, which consists of the following four steps. (1) SCE first evaluates the initial accuracy of the LLM on a dataset $\mathcal{X}$, denoted as *Acc*, and selects the subset that is predicted correctly:

$$\mathcal{X}_s = \{x_i \in \mathcal{X} \mid LLM : x_i \mapsto y_i\}, \tag{3}$$

where $\mathcal{X}_s$ denotes the selected subset. (2) For each selected instance $x_i \in \mathcal{X}_s$, SCE first identifies its set of irrelevant labels $\hat{\mathcal{Y}}_i = \{\hat{y}_i \in \mathcal{Y} \mid \hat{y}_i \neq y_i\}$. Then, for each $\hat{y}_i$, LCD is employed to discover biased words that drive the LLM to generate the irrelevant label:

$$\mathcal{W}_i = \{LCD(\hat{y}_i) \mid \hat{y}_i \in \hat{\mathcal{Y}}_i\}, \tag{4}$$

where $|LCD(\hat{y}_i)| = m$, denoting that the number of biased words is limited to m, $\mathcal{W}_i$ denotes all discovered biased words for x_i. As shown in Fig. 3b, LCD takes the irrelevant label *Instrument-Agency* as input and utilizes causal discovery to identify a biased word *Probe* that drives the LLM to generate *Instrument*.

(3) SCE then inserts each discovered biased word into the original sentence individually to construct noisy data:

$$\mathcal{X}'_i = \{(x'_i = (s_i \oplus w_i, e_i), y_i) \mid w_i \in \mathcal{W}_i\}, \tag{5}$$

where $s_i \oplus w_i$ denotes that a biased word w_i is inserted into the sentence s_i, and $\mathcal{X}'_i$ represents all noisy data constructed from x_i. Insertion is a commonly used perturbation approach for robustness evaluation (Hu et al., 2025; Li et al., 2019; Ribeiro et al., 2020), and mini-

mal insertion helps avoid introducing additional spurious correlations. To ensure task-level semantic consistency (Ribeiro et al., 2020), we introduce the following insertion strategies. For context-based tasks like RE, a biased word is inserted as a modifier for a given entity, as shown in Fig. 3b. For entity-centric tasks like ET, a biased word is inserted as a modifier for another noun in the sentence. A detailed explanation with examples is provided in Sect. 6.3.

(4) Finally, SCE evaluates robustness using the noisy set $\mathcal{X}' = \{\mathcal{X}'_i \mid x_i \in \mathcal{X}_s\}$. As shown in Fig. 3b, the LLM is misled by the spurious correlation induced by the biased word *Probe*, explicitly exposing the lack of deep semantic understanding.

4.4 Evaluation Metrics

To quantify the robustness, we first introduce the instability hit function $@n(\mathcal{X}'_i)$ to assess whether the LLM exhibits instability on an instance x_i:

$$@n(\mathcal{X}'_i) = \begin{cases} 1, & if\ error(\mathcal{X}'_i) \geq n \\ 0, & else \end{cases}, \tag{6}$$

where $error(\mathcal{X}'_i)$ denotes the number of prediction errors in the noisy data $\mathcal{X}'_i$, and n is a predefined threshold for instability hits. On this basis, the robustness is evaluated by:

$$Ins@n = \frac{\sum_{\mathcal{X}'_i \in \mathcal{X}'} @n(\mathcal{X}'_i)}{|\mathcal{X}'|}, \tag{7}$$

where the instability ratio $Ins@n$ denotes the proportion of unstable instances; a smaller n indicates a stricter criterion. Based on this, we propose an accuracy metric:

$$Acc_s[n] = Acc - Acc \cdot Ins@n, \tag{8}$$

where $Acc_s[n]$ represents stable accuracy, which integrates initial and robust performance to reflect the model's genuine capability in real-world scenarios.

4.5 Explanation of LCD

4.5.1 Pipeline of LCD

LCD takes an irrelevant label $\hat{y}_i$ as input and identifies the corresponding biased words, which consists of two steps, as depicted in Fig. 4. Note that each word in $\hat{y}_i$ is processed separately. We present only one case, denoted as $\hat{y}_i^j$, which indicates the j-th word in $\hat{y}_i$. (1) LCD retrieves ConceptNet (Speer et al., 2017) for words related to $\hat{y}_i^j$, specifically, those connected via *RelatedTo* edges:

$$\mathcal{R}_i = \{r_i \mid (r_i, RelatedTo, \hat{y}_i^j) \in ConceptNet\}, \tag{9}$$

where r_i denotes a related word and $\mathcal{R}_i$ indicates the full set. This setup narrows the search space, as related words are more likely to be the causes for the generation of $\hat{y}_i^j$. To facilitate subsequent processing, we pair them up:

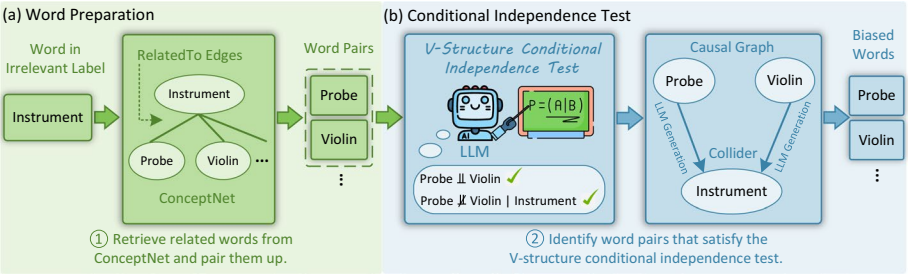

Fig. 4 Pipeline of the LCD module. **a** Each node in ConceptNet represents a concept word. **b** Each word is treated as a variable in the conditional independence test without context

$$\mathcal{T}_i = \{(r_i^h, r_i^t) \mid (r_i^h, r_i^t \in \mathcal{R}_i) \wedge (r_i^h \neq r_i^t)\}, \tag{10}$$

where (r_i^h, r_i^t) indicates a word pair, including head and tail words. This stage prepares words that serve as candidates for biased words, as shown in Fig. 4a.

(2) For each word pair, LCD assumes them to be the parents in a V-structure, sets $\hat{y}_i^j$ as the collider, and employs the evaluated LLM to conduct the conditional independence test. If the independencies are satisfied, the V-structure is inferred to exist. As a result, two causes of generating $\hat{y}_i^j$ are identified in the absence of context and are thus regarded as biased words. As shown in Fig. 4b, the LLM determines that *Probe* $\perp\!\!\!\perp$ *Violin* and *Probe* $\not\perp\!\!\!\perp$ *Violin* *Instrument*, indicating the *Probe* $\rightarrow$ *Instrument* $\leftarrow$ *Violin* structure holds, where *Probe* and *Violin* serve as biased words. LCD repeats the process for all $\hat{y}_i$:

$$\mathcal{W}_i = \{r_i^h, r_i^t \mid (r_i^h \perp\!\!\!\perp r_i^t) \wedge (r_i^h \not\perp\!\!\!\perp r_i^t \mid \hat{y}_i^j)\}, \tag{11}$$

where $(r_i^h, r_i^t) \in \mathcal{T}_i$, $\hat{y}_i^j \in \{\hat{y}_i \in \mathcal{Y} \mid \hat{y}_i \neq y_i\}$. The notation $r_i^h \perp\!\!\!\perp r_i^t$ indicates that head and tail words are statistically independent, while $r_i^h \not\perp\!\!\!\perp r_i^t \mid \hat{y}_i^j$ implies that they become statistically dependent in the condition of $\hat{y}_i^j$. $\mathcal{W}_i$ represents all discovered biased words for the instance x_i. LLMs perform this independence test through their internal parameters, which will be introduced in detail later.

4.5.2 LLM-Based Conditional Independence Test

This section describes how the conditional independence test is performed using LLMs. Obviously, LLMs cannot calculate the equalities in Eqs. 1 and 2. Therefore, we develop a set of criteria aligned with their autoregressive nature (Radford et al., 2019). For Eq. 1, which tests independence, we propose the following adapted criterion:

$$P(A \mid B) \leq P(A) + \epsilon \Rightarrow A \perp\!\!\!\perp B, \tag{12}$$

$$P(A \mid B) > P(A) + \epsilon \Rightarrow A \not\perp\!\!\!\perp B, \tag{13}$$

where ϵ denotes the error tolerance. Since the self-attention mechanism encodes global dependencies (Vaswani et al., 2017), variables in LLMs are never strictly independent.

Thus, we adopt an approximate estimation. Equation 12 indicates that given word B, the probability of generating next-word A is at most $P(A) + \epsilon$, the baseline without any prior information. This implies that B offers no cue for A, and thus A and B are statistically independent. Implicit premise: $P(A)$ is very small, and $P(A \mid B)$ remains so if A and B are independent. Note that ϵ is added to $P(A)$ since it is relatively low without any prompt, and $P(A \mid B)$ must significantly exceed $P(A)$ to indicate dependency. Consider the example in Fig. 4b: if the LLM estimates that $P(Probe \mid Violin) \leq P(Probe) + \epsilon$, the two words are deemed statistically independent for it, i.e., *Probe* $\perp\!\!\!\perp$ *Violin*.

For Eq. 2, which involves conditional independence, further adjustment is required, as the autoregressive nature of LLMs (Radford et al., 2019) prevents them from computing joint probabilities.[2] Such as the joint condition B and C in $P(A \mid B, C)$. To address this challenge, we introduce a revised testing criterion with a new formulation:

$$P(A\,|\,B\,|\,C) < P(A\,|\,C) - \epsilon \Rightarrow A \perp\!\!\!\perp B\,|\,C, \tag{14}$$

$$\begin{aligned} P(A\,|\,B\,|\,C) \geq P(A\,|\,C) - \epsilon \Rightarrow A \not\perp\!\!\!\perp B\,|\,C, \\ \text{s.t.} \quad A \not\perp\!\!\!\perp C, \end{aligned} \tag{15}$$

where $P(A\,|\,B\,|\,C)$, a formulation introduced in this work, represents the sequential generation process of LLMs: given word C, LLMs generate A and B in sequence. With C denotes a collider, we assume by default that $A \not\perp\!\!\!\perp C$. Based on this, Eq. 15 indicates that C is known to increase the probability of the LLM generating A; when B is inserted, this probability is roughly maintained or even increases, suggesting that A and B exhibit statistical dependence in the condition of C. Here, the "roughly maintained" means allowing a slight reduction, i.e., minus ϵ from $P(A \mid C)$. Consider the example in Fig. 4b, if the LLM estimates that $P(Probe \mid Violin \mid Instrument) \geq P(Probe \mid Instrument) - \epsilon$, *Probe* and *Violin* are regarded as statistically dependent under the condition of *Instrument* for the LLM, i.e., *Probe* $\not\perp\!\!\!\perp$ *Violin* *Instrument*. This example aligns with our intuition: *Probe* and *Violin* are typically unrelated, but can be connected through the attribute *Instrument*.

4.5.3 LLM-Based Probability Estimation

To estimate probabilities for the above equations, we propose an LLM-based probability estimation approach, which integrates prompts and model logits, as shown in Fig. 5. Specifically, we define four probability prompts corresponding to the probabilistic expressions in these equations. To ensure reliability and reduce interference, we follow two principles: (1) retain only essential words and (2) ensure lexical consistency across all prompts, as shown in Fig. 5a. During estimation, words are inserted into the prompts and their probabilities are extracted from the output logits of LLMs.

For clarity, we use the estimation of $P(A = a \,|\, B = b)$ as an example. We first obtain the probability distribution at the position in the condition of $B = b$:

$$P(A \mid B = b) = LLM(pmt_2(b))[pos_A^s - 1], \tag{16}$$

[2] A proof by contradiction is presented in Appendix A.1.

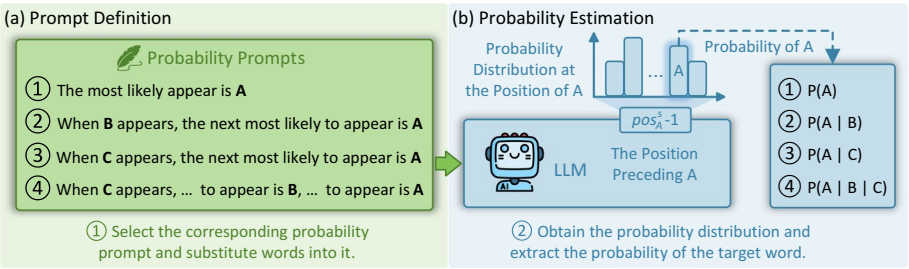

Fig. 5 The pipeline of the LLM-based probability estimation. **a** Each prompt corresponds to its probabilistic expression via the matching indices. **b** pos_A^s indicates the position of A in the sequence

Table 1 Statistics of the datasets used in experiments

Task	Dataset	Size	# Labels	# Words
ED	ACE 2005	2494	18	31
ET	Few-NERD	2800	14	14
RE	SemEval	2263	9	17

where pmt_2 represents the second prompt in Fig. 5a, b denotes the given word substituted in B, and pos_A^s indicates the position of A in the sequence. $pos_A^s - 1$ means that the probability distribution at A's position comes from preceding outputs. Then, we extract the probability of word a from the distribution, as depicted in Fig. 5b:

$$P(A = a \mid B = b) = P(A \mid B = b)[pos_a^v], \tag{17}$$

where pos_a^v denotes the position of a in vocabulary. Such probability estimates enable the previous conditional independence test and discover biased words.

5 Experiments

5.1 Datasets

We select three commonly applied IE tasks, including ED, ET, and RE. Specifically, ED identifies the event type triggered by a given entity; ET determines the category of a given entity; and RE predicts the relation between a pair of given entities. These tasks target semantic knowledge at the factual, categorical, and relational levels, respectively. The word-rich labels in these tasks enable the creation of diverse spurious correlations, facilitating extensive evaluation. We instantiate these tasks on three datasets: ACE 2005 (Grishman et al., 2005) for ED, Few-NERD (Ding et al., 2021) for ET, and SemEval (Hendrickx et al., 2010) for RE. The statistics are presented in Table 1. More details are provided in Appendix A.2.

Specifically, for ACE 2005, we select 18 event types and cap the number of instances at 200 per type. For Few-NERD, we choose 14 entity types, also with 200 instances each. For SemEval, we use the test set and exclude instances labeled as *Other*, as this label lacks specific semantic meaning. Notably, SemEval relations are bi-directional; for simplicity, we

ignore directionality. In both ACE 2005 and SemEval, a label encompasses multiple words. We process each word separately to construct noisy data.

5.2 Evaluated LLMs

Regarding feasibility, we select two SOTA open-source LLM families: the Llama (Dubey et al., 2024) and Qwen (Yang et al., 2025) series.[3]. From the Llama family, we include Llama-3, Llama-3.1, and the latest Llama-3.3; from Qwen, we select Qwen-1.5, Qwen-2, Qwen-2.5, and the recently released Qwen-3. These models span a range of parameter sizes, from approximately 7B to 70B, enabling us to examine how robustness evolves across different parameters and dataset sizes. Recent reports indicate that Llama-3-70B surpasses early versions of GPT-4 (Achiam et al., 2023) on various semantic understanding tasks (Dubey et al., 2024). Moreover, Qwen-3-32B outperforms the recent GPT-4o-mini (Hurst et al., 2024) in semantic understanding and also exhibits stronger reasoning ability than DeepSeek-R1-Distill-Llama-70B (Guo et al., 2025) under thinking mode (Yang et al., 2025). In summary, they catch up with closed-source ones.

For closed-source models, we include GPT-4o (Hurst et al., 2024), the current top-performing model, along with its lightweight versions, GPT-4o-mini and GPT-4.1-mini. Although their logits are inaccessible, we can still evaluate their robustness based on biased words identified from open-source models as an approximate estimation.

5.3 Robustness Evaluation Baselines

We validate the effectiveness of SCE through comprehensive comparisons with classical methods and two categories of LLM evaluation approaches: classical-based and LLM-specific. The classical baselines: **StressTest** (Naik et al., 2018) randomly swaps adjacent characters within words and inserts irrelevant phrases as distractions. **PWWS** (Ren et al., 2019) computes word saliency and iteratively replaces words with synonyms based on saliency ranking. **TextBugger** (Li et al., 2019) identifies important words and generates perturbation bugs via insertion, deletion, swapping, and substitution. **TextFooler** (Jin et al., 2020) ranks words by importance and iteratively replaces them with semantically filtered synonyms. **CheckList** (Ribeiro et al., 2020) swaps adjacent characters and inserts randomly generated URLs.

Classical-based baselines: **Prompt-Robust** (Zhu et al., 2023) incorporates StressTest, TextBugger, TextFooler, and CheckList for an integrated evaluation. **RUPBench** (Wang & Zhao, 2024) applies diverse lexical and syntactic perturbations, such as character swapping, structural rephrasing, and distraction techniques from StressTest and CheckList. **ABFS** (Xiao et al., 2025) uses the Best-First Search algorithm to identify synonym replacements that most significantly affect model confidence scores.

LLM-specific baselines: **NEO-BENCH** (Zheng et al., 2024) evaluates LLM robustness against distribution drifts by introducing neologisms. **ToxiCloakCN** (Xiao et al., 2024) studies the effect of introducing emojis on LLM robustness. **J&H** (Hu et al., 2025) employs synonym replacement, irrelevant phrase insertion, and semantically confusable sentences to challenge LLMs.

[3] The top 10 models on the "Open LLM Leaderboard" are all based on either Llama or Qwen models.

5.4 Implementation Details

Noisy data construction. To balance computational cost and validation sufficiency, we select 10 biased words per word associated with irrelevant labels for each instance, i.e., $m = 10$ in Sect. 4.3. The error tolerance ϵ in Sect. 4.5.2 is defined by the difference in order of magnitude between two values and set to 0.3. For example, with a tolerance of $+\epsilon$, a probability of 0.08 is considered approximately equal to 0.12, but not to 0.16. **LLM robustness evaluation**. We utilize in-context learning (Dong et al., 2022) with two manually crafted examples to guide LLMs in task execution. Detailed prompts are provided in Appendix A.3. To ensure effective in-context learning, all LLMs are instruction-tuned versions. Following prior works (Jin et al., 2024a, b), we set the temperature to 0 for stable inference. Robustness and actual performance is quantified by the instability ratio *Ins@n* and stable accuracy $Acc_s[n]$ ($n = 1, 2, 3$), respectively, as defined in Sect. 4.3. To improve computational efficiency, we perform parallel inference using the vLLM[4] framework. To reduce computational cost, we evaluate the closed-source models using 10% random sampling. **Baseline reproduction**. For fairness, we generate the same amount of noisy data for all baselines as for SCE. To maintain label consistency, we exclude perturbations that may alter the original meaning of instances, such as homophone replacement (Wang & Zhao, 2024; Xiao et al., 2024). The 72B model was quantized using a 4-bit method, while all other models were quantized to 8-bit for efficiency on A800 GPUs.

5.5 Main Results

The main results are presented in Table 2, from which we summarize the following three findings. (1) **LLMs are notably fragile to word-level spurious correlations**. The color indicator intuitively reflects the marked fragility across all evaluated LLMs. Even the SOTA models Qwen-2.5-72B and Qwen3-32B show instability ratios as high as 30.78% and 30.34% in RE, respectively. In addition, lightweight LLMs that are widely applied in the research community (Hu et al., 2025; Xiao et al., 2024; Zheng et al., 2024) face greater challenges, as the instability ratio of Qwen-2-7B exceeds 50%. These results rigorously verify the observation of the spurious correlation phenomenon in IE tasks (Zhang et al., 2024). This suggests that LLMs lack an in-depth semantic understanding, revealing their significant reliance on statistical patterns. This flaw damages their actual performance, as shown in Fig. 6, where their stable accuracy drops significantly compared to their initial accuracy. For example, the average performance of Qwen-3-32B drops from 81.81 to 63.0%. Even enabling its reasoning mode, i.e., Qwen-3-32B-Thinking, proves ineffective and even aggravates the issue, as reasoning may expose additional spurious correlations. This reflects the severe risks of LLMs in IE applications, particularly in multi-entity task RE, which demands richer information and encounters the issue of relation overlap (Sun et al., 2020).

(2) **Accuracy can be misleading under spurious correlations**. As shown in Fig. 7, a negative correlation is observed between accuracy and stable accuracy, implying that models with higher accuracy might be less robust. For example, although Qwen-2-7B outperforms Llama-3-70B by 10.96% in accuracy, its stable accuracy is 11.93% lower. A similar pattern is observed among the Llama models as well. This highlights that traditional metrics may be inadequate for evaluating LLMs, as data contamination becomes a prevalent

[4] https://docs.vllm.ai/en/latest/.

Table 2 Main results (%). Acc: initial accuracy; $Ins@1$: instability ratio; $Acc_s[1]$: stable accuracy ($n = 1$)

Model	Event Detection			Entity Typing			Relation Extraction		
	Acc	$Ins@1 \downarrow$	$Acc_s[1]$	Acc	$Ins@1 \downarrow$	$Acc_s[1]$	Acc	$Ins@1 \downarrow$	$Acc_s[1]$
Llama-3-8B	66.88	25.24	50.00	80.96	25.89	60.00	53.91	67.62	17.45
Llama-3-70B	84.84	13.37	73.50	82.18	11.86	72.43	57.67	21.99	44.98
Llama-3.1-8B	79.71	19.42	64.23	81.25	10.20	72.96	65.89	53.92	30.36
Llama-3.1-70B	84.12	17.11	69.73	**85.71**	17.71	70.54	62.31	28.58	44.50
Llama-3.3-70B	87.73	11.88	77.31	84.25	15.30	71.36	70.22	23.10	**54.00**
Qwen-1.5-7B	73.06	41.77	42.54	52.75	51.05	25.82	45.16	60.67	17.76
Qwen-1.5-32B	83.88	22.04	65.40	79.82	16.87	66.36	65.89	30.32	45.91
Qwen-1.5-72B	85.85	16.95	71.29	81.61	17.37	67.43	72.43	29.59	50.99
Qwen-2-7B	80.83	35.42	52.21	82.46	21.61	64.64	68.63	51.84	33.05
Qwen-2-72B	88.81	13.63	76.70	84.54	14.20	72.54	74.33	30.26	51.83
Qwen-2.5-7B	79.15	37.28	49.64	81.82	26.58	60.07	67.43	36.83	42.60
Qwen-2.5-32B	87.89	16.15	73.70	84.96	13.87	**73.18**	70.97	31.01	48.96
Qwen-2.5-72B	**90.18**	10.09	**81.07**	84.36	15.66	71.14	77.24	30.78	53.47
Qwen-3-14B	86.57	15.61	73.06	83.25	18.70	67.68	60.10	38.24	37.12
Qwen-3-32B	87.77	19.28	70.85	83.96	20.46	66.79	73.71	30.34	51.35
Qwen-3-32B-Thinking	88.65	41.18	52.15	83.39	29.18	59.06	**79.10**	35.20	51.26

Results of $n = 2, 3$ are provided in Appendix B.1. **Bold** and underlined denote the best and second-best results, respectively. Darker indicates greater instability

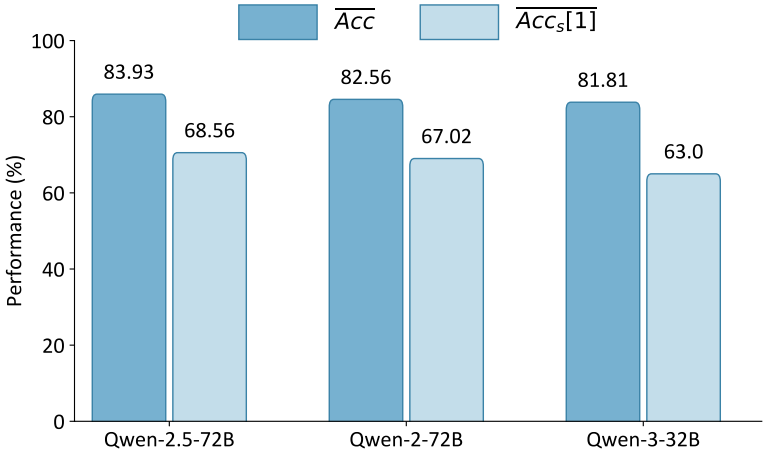

Fig. 6 Comparison between average accuracy ($\overline{Acc}$) and average stable accuracy ($\overline{Acc_s[1]}$)

issue (Deng et al., 2024; Li et al., 2024). While accuracy primarily assesses knowledge memorization rather than understanding. Therefore, our stable accuracy serves as a necessary complement. This shows that LLMs may perform far from expectations, offering a unique perspective for model selection in risk management.

(3) **Scaling laws face a bottleneck in handling spurious correlations**. For clarity, we analyze scaling laws for model parameters and dataset size separately. For model parameters, as shown in Fig. 8a, we observe that scaling model parameters from 7B to 32B significantly enhances the robustness across all tasks. However, further scaling to 72B parameters results in a plateau or even a slight robustness decline. This indicates that parameter scal-

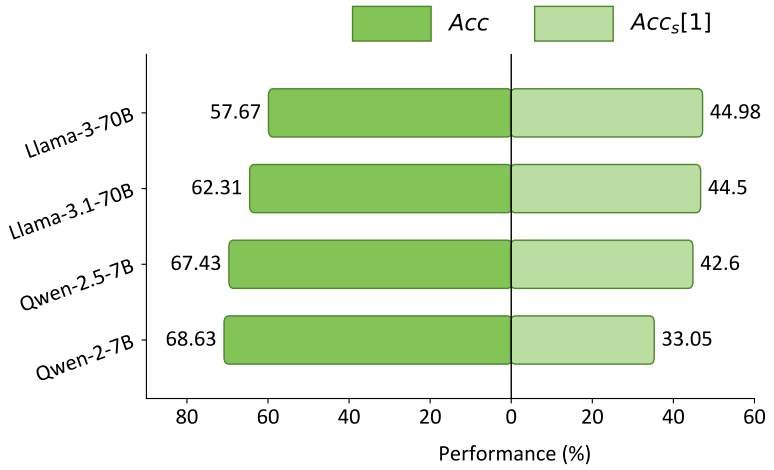

Fig. 7 Negative correlation between accuracy (Acc) and stable accuracy ($Acc_s[1]$) in RE

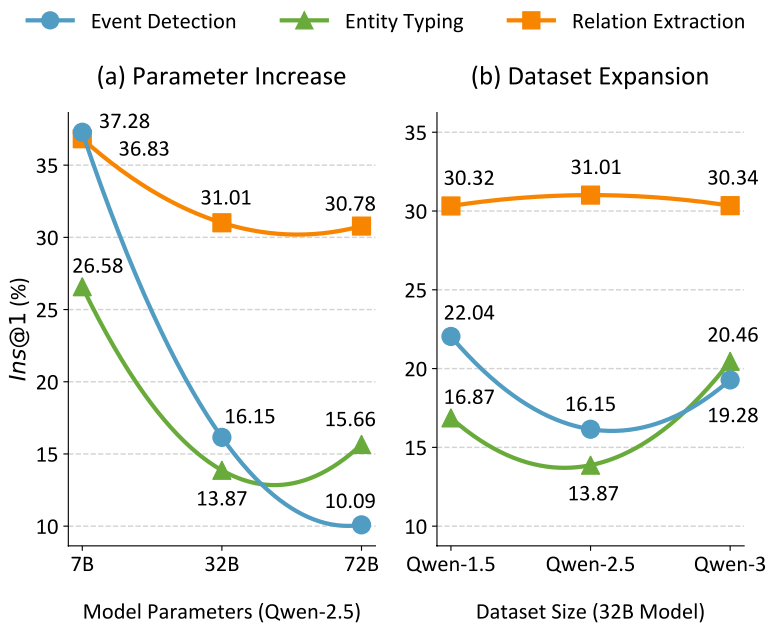

Fig. 8 Instability trends under scaling with **a** increasing parameters and **b** expanding dataset

ing enhances robustness, but the benefit is limited. For dataset size, as shown in Fig. 8b, we observe that from Qwen-1.5 to Qwen-2.5, as researchers collect a significantly larger volume of data (Hui et al., 2024), the robustness is obviously enhanced in both ED and ET. Unexpectedly, the more extensive dataset in Qwen-3 (Yang et al., 2025) results in a higher instability. Even worse, the instability ratio in RE stays consistently high, despite continued improvements in accuracy, as shown in Table 2. We suspect this is because human knowl-

edge has been exhausted (Villalobos et al., 2024), and more data merely repeats existing information. At this stage, perhaps we should prioritize data quality (Wu et al., 2025). This confirms that scaling alone cannot eliminate spurious correlations, which are likely to persist as LLMs evolve. SCE offers a tool for continuous monitoring.

5.6 Comparison Results

The comparison results are shown in Table 3. We report the instability ratio, where higher values indicate greater perturbation performance. The conclusions can be summarized as follows. (1) SCE achieves a significant performance lead over all baselines. For example, compared to the second-best method, NEO-BENCH (Zheng et al., 2024), SCE improves the average instability ratio by 20.96% on Llama-3-8B and 15.66% on Qwen-2-7B.Its superior efficiency stems from the induction of spurious correlations, confirming that they are still the key source of instability (Cui & Athey, 2022; Ye et al., 2024). (2) Although NEO-BENCH and ToxiCloakCN (Xiao et al., 2024) introduce distractors such as neologisms or emojis, their perturbation effectiveness remains limited, as LLMs may disregard features that deviate from the training distribution. In contrast, we avoid this issue by only utilizing basic words. (3) Classical methods exhibit relatively weak perturbation effects, as LLMs are already robust to common perturbations (Singh et al., 2024) such as synonym replacement. Thus, classical-based methods, such as PromptRobust (Zhu et al., 2023) and RUP-Bench (Wang & Zhao, 2024), also exhibit limited performance. This suggests that SCE can

Table 3 Comparison results of *Ins*@1 (%)

Type	Method	Llama-3-8B				Qwen-2-7B			
		ED	ET	RE	Avg	ED	ET	RE	Avg
Classical	StressTest (Naik et al., 2018)	11.80	11.06	20.15	14.34	9.72	6.59	14.15	10.15
	PWWS (Ren et al., 2019)	5.39	6.84	9.75	7.33	6.89	5.16	8.07	6.71
	TextBugger (Li et al., 2019)	11.98	12.13	20.23	14.78	11.50	8.15	14.85	11.50
	TextFooler (Jin et al., 2020)	10.25	10.84	15.56	12.22	10.61	7.71	12.16	10.16
	CheckList (Ribeiro et al., 2020)	11.56	10.57	22.52	14.88	10.76	7.15	14.02	10.64
Classical-Based	PromptRobust (Zhu et al., 2023)	15.40	<u>14.48</u>	26.62	18.83	14.92	10.75	19.85	15.17
	RUPBench (Wang & Zhao, 2024)	16.06	12.57	<u>29.65</u>	<u>19.43</u>	14.92	8.28	21.64	14.95
	ABFS (Xiao et al., 2025)	8.45	8.88	14.41	10.58	8.97	6.85	11.27	9.03
LLM-Specific	NEO-BENCH (Zheng et al., 2024)	<u>16.48</u>	11.11	28.26	18.62	<u>19.58</u>	<u>13.13</u>	<u>29.19</u>	<u>20.63</u>
	ToxiCloakCN (Xiao et al., 2024)	9.23	7.60	17.36	11.40	9.12	8.32	19.72	12.39
	J&H (Hu et al., 2025)	14.26	14.04	29.32	19.21	17.40	10.40	28.30	18.70
	SCE (Ours)†	**25.24**	**25.89**	**67.62**	**39.58**	**35.42**	**21.61**	**51.84**	**36.29**

Results of $n = 2, 3$ are in Appendix B.2. **Bold** and <u>underlined</u> denote the best and second-best perturbation performance, respectively. † indicates a statistically significant difference from the second-best group ($p < 0.05$)

serve as a diagnostic framework for long-term stability evaluation, as spurious correlations are inherent to LLMs.

5.7 Ablation Study

To evaluate the effectiveness of LCD, we perform an ablation study by replacing its causal discovery techniques with random sampling. Specifically, we sample from the RelatedTo set within ConceptNet, ConceptNet, and English vocabulary, as shown in Table 4. From these results, we derive the following conclusions: (1) LCD effectively identifies biased words or spurious correlations, as it significantly increases the instability. (2) LCD significantly outperforms RandomRelatedTo, indicating that not all related words to irrelevant labels function as biased words, and our causal discovery method can effectively identify them. (3) RandomRelatedTo outperforms the other two sampling methods, revealing that related words more easily induce potential spurious correlations, which also explains why it outperforms the baselines in Table 3.

6 In-Depth Analysis

6.1 Data Quality Assessment

To evaluate the quality of noisy data, we assess it from three aspects: grammar, semantics, and label consistency. Specifically, we randomly sampled 50 instances from each dataset. For grammar, we use Grammarly[5] measure the error growth. For semantics, we measure the semantic similarity between the original and the noisy sentences using Sentence-BERT (Reimers & Gurevych, 2019). As shown in Fig. 9, over 90% of the noisy data maintains high quality in both grammar and semantics, largely benefiting from the minimal perturbation of inserting only one word. In addition, we manually inspect the labels with three domain-specific annotators and confirm that all labels remain identifiable and contextually consistent after perturbation. Table 5 provides noisy data for reference.

Table 4 Ablation results of *Ins*@1 (%)

Method	Llama-3-8B				Qwen-2-7B			
	ED	ET	RE	Avg.	ED	ET	RE	Avg.
SCE w/ LCD†	**25.24**	**25.89**	**67.62**	**39.58**	**35.42**	**21.61**	**51.84**	**36.29**
LCD → RandomRelatedTo	<u>16.43</u>	<u>13.15</u>	<u>33.03</u>	<u>20.87</u>	<u>21.73</u>	<u>12.67</u>	<u>38.76</u>	<u>24.39</u>
LCD → RandomConceptNet	14.63	9.93	29.84	18.13	16.62	11.26	31.62	19.83
LCD → RandomEnglishWord	15.23	10.76	29.84	18.61	17.36	10.65	30.59	19.53

Results of $n = 2, 3$ are in Appendix B.3

Bold/<u>underlined</u>: the best/second-best perturbation performance.†: significant difference from the second-best group ($p < 0.05$). w/: "with". →: replacing LCD with random sampling in a specified space

[5] https://app.grammarly.com/.

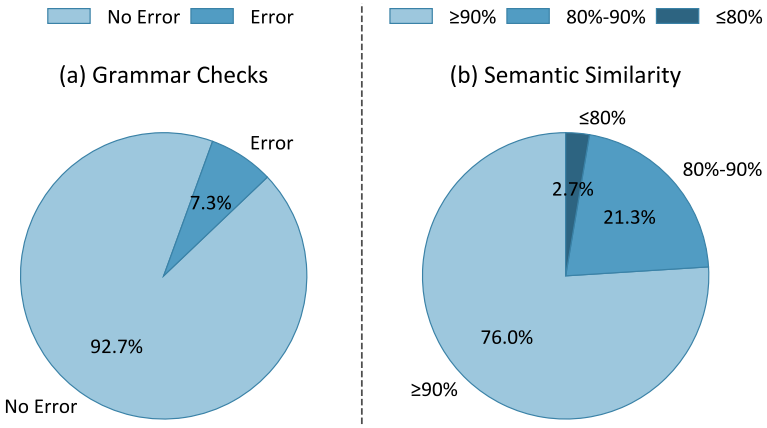

Fig. 9 Results of data quality assessment, including **a** grammar checks and **b** semantic similarity

Table 5 Case study from Qwen-2-7B on three evaluated tasks

Case	Noisy sentence	Label	Prediction
ED-1	... they **fire** adisagreementguy for his political views	Personnel:End-Position	Conflict:Attack
ED-2	... as a **former** federalindictmentprosecutor, every ..	Personnel:End-Position	Justice:Charge-Indict
ET-1	The cast included **Cyril Raymond**,aviator Aubrey ..	Actor	Airplane
ET-2	The armament of the **Type 2 Ho-I** was abulkheadgun	Weapon	Ship
RE-1	... requires a text **note** to explain thewholeness **rating**	Message-Topic	Component-Whole
RE-2	**Water** evaporated from thegovernment **evaporators** ..	Entity-Origin	Instrument-Agency

Bold denotes the given entities, whileunderlinedindicates the inserted biased words. Sentences are abbreviated for clarity

6.2 Conditional Independence Assessment

Furthermore, to evaluate the rationality of our conditional independence testing techniques in Sect. 4.5.2, we manually annotated an assessment set. Specifically, we first search Wikipedia[6] for statistically correlated word pairs. For example, *Confounding* co-occurs with *Causal Inference* in the same conceptual explanation, hence we label them as *Confounding* $\not\perp\!\!\!\perp$ *Causal Inference* ($A \not\perp\!\!\!\perp B$). Conversely, we empirically select a word unrelated to *Confounding*, such as *Literature*, and annotate them as *Confounding* $\perp\!\!\!\perp$ *Literature* ($A \perp\!\!\!\perp B$). At this stage, we have constructed the test set for independence. Building on this, we add appropriate conditions to form the conditional independence test set. Firstly, for $A \not\perp\!\!\!\perp B$, to preserve their dependence, we select a condition correlated to both words. For example, we select *Causal Study*, which co-occurs to both *Confounding* and *Causal Inference* in the same page, yielding *Confounding* $\not\perp\!\!\!\perp$ *Causal Inference* *Causal Study* ($A \not\perp\!\!\!\perp B \mid C$). Secondly,

[6] https://www.wikipedia.org/.

for $A \perp\!\!\!\perp B$, to avoid introducing any association between A and B, we need a condition related to either A or B, but not both. Coincidentally, *Causal Study* is correlated to *Confounding* but not to *Literature*, thereby forming *Confounding* $\perp\!\!\!\perp$ *Literature* $\not\perp\!\!\!\perp$ *Causal Study* ($A \perp\!\!\!\perp B \mid C$). Repeating the process yields an assessment set of 100 instances.

The assessment results are shown in Fig. 10. Across all four conditional independence criteria, based on our techniques, LLM judgments exhibit strong agreement with human annotations, all exceeding 80% and most surpassing 90%. The results validate the validity of our tailored criteria and the probability estimation method, providing a solid foundation for reliable causal discovery. Full instances of the assessment set and detailed prediction results of the LLM are available in Appendix B.4.

6.3 Case Study

For clarity, a case study is provided in Table 5. We first use these examples to illustrate the insertion rule introduced in Sect. 4.3, followed by a detailed analysis of how biased words mislead LLMs. For the insertion strategy, each biased word serves as a noun modifier according to the following two rules. (1) In entity-centric tasks, such as ED and ET, the biased word is inserted as a modifier to the following noun to avoid altering the given entity's inherent properties. For example, in case ED-1, the inserted biased word *disagreement* acts as a modifier to the following noun *guy*, ensuring that the event type of *fire* remains unaffected. (2) In context-based tasks such as RE, where relations depend on context (Zhang et al., 2023), the biased word is inserted as a modifier to an entity. For example, in case RE-1, the biased word *wholeness* modifies the entity *rating*, without influencing how *explain* contributes to the relation. Minor grammatical errors in the noisy data are acceptable, as the task semantics remain unchanged (Ribeiro et al., 2020).

These examples provide an intuitive illustration of how biased words mislead LLMs. For instance, in case RE-2, the correct relation between *water* and *evaporators* is *Entity-Origin*. However, a biased word *government* could drive an LLM to generate word *Agency* regardless of context, corresponding to the irrelevant label *Instrument-Agency*, and it is inserted into the noisy sentence, thereby inducing a spurious correlation between the noisy sentence and *Instrument-Agency*. The LLM exploits the spurious correlation as a shortcut, ignoring the contextual semantics of *government*, leading to the incorrect prediction. In other exam-

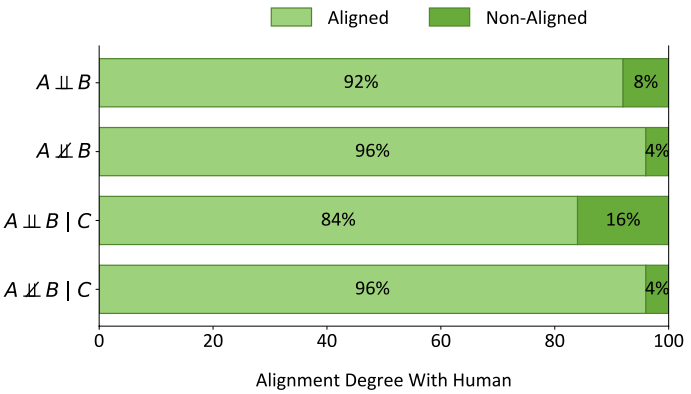

Fig. 10 Results of conditional independence assessment on Qwen-2-7B with manual annotations

ples, the generative relationship between biased words and irrelevant labels is as follows: *disagreement* → *Conflict*, *indictment* → *Charge*, *aviator* → *Airplane*, *bulkhead* → *Ship*, and *wholeness* → *Whole*.

6.4 Closed-Source Model Study

To evaluate the robustness of commercial closed-source LLMs, we conduct experiments on the widely applied GPT-4o, GPT-4o-mini (Hurst et al., 2024), and GPT-4.1-mini. Although their logits are not accessible, we transfer the biased words identified in Qwen-2-7B to these models, leveraging their similarity in training data (Cheng et al., 2025) and model architectures (Vaswani et al., 2017). Specifically, we first evaluate them via the API on the full dataset, then randomly sample 10% of the correctly predicted instances for perturbation evaluation to reduce computation. The results are presented in Fig. 11, from which we draw the following observations. (1) Even SOTA commercial LLMs are fragile to word-level spurious correlations. For example, despite GPT-4o achieving an impressive accuracy of 80.6% in RE, which surpasses all open-source models, its stable accuracy declines to 65.95% under spurious correlation interference. Recent studies have also found that GPT-4o is unstable (Shahriar et al., 2024; Yu et al., 2025), but we are the first to explore it from a causal perspective. (2) Lightweight models exhibit greater fragility, and closed-source models are no exception. For example, GPT-4o-mini and GPT-4.1-mini experience significant drops in stable accuracy, with decreases of 42.67% and 50.34% respectively. These declines are close to those observed in open-source models Qwen-2.5-7B and Qwen-2-7B.

6.5 Robustness Enhancement Study

This section investigates how SCE evaluation results can be leveraged as feedback to improve robustness. Our findings demonstrate that counterfactual (CF) examples (Miao et al., 2023, 2024; Zhang et al., 2023) derived from SCE feedback effectively mitigate the impact of spurious correlations. Specifically, we focus on RE and begin with an error analysis following SCE evaluation. As shown in Fig. 12, Llama-3-8B makes the most errors

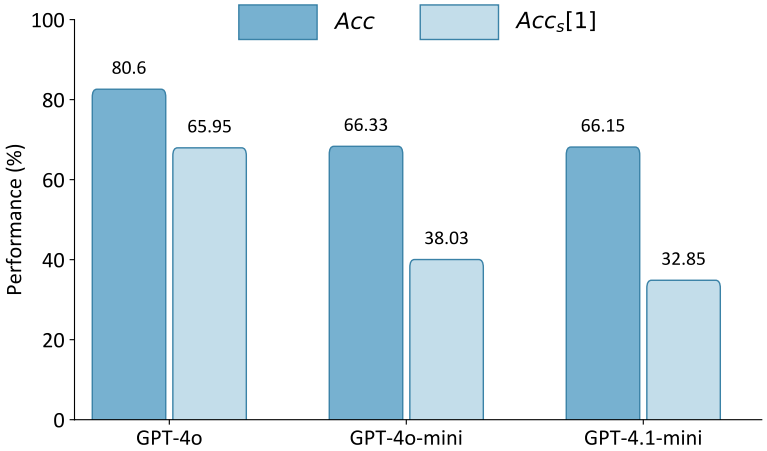

Fig. 11 Accuracy (Acc) vs. stable accuracy ($Acc_s[1]$) of closed-source LLMs on RE

between *Entity-Origin* and *Cause-Effect* relations, revealing the hotspots of spurious correlations within RE. Based on this observation, we regard it as prior knowledge and enhance model robustness through the following strategies: (1) **CoT**: prompting a model to reason step by step. (2) **R1-Instruction**: prompting a model to pay attention to the most error-prone relation pair (R1). (3) **R1-Example**: creating in-context examples based on the relations in R1. (4) **R1-CF**: constructing counterfactual in-context examples between the relation pair R1 (refer to Table 9). (5) **R2/3-CF**: applying the second/third most error-prone relation pair to create counterfactuals. In addition, we report a **w/o R1-Label** setting, where the relations in R1 are excluded from evaluation results. To ensure soundness, R1/2/3 are identified on the test set but used for the validation set.

The results are presented in Table 6, leading to the following conclusions. (1) The integration of SCE and counterfactuals yields the best performance, as R1-CF brings the superior gain. For example, the average stable accuracy of Llama-3-8B is improved by 5.38%, while maintaining a minimal average instability ratio. We attribute this to SCE's effectiveness in diagnosing model weaknesses, which counterfactuals efficiently help address by clarifying the decision boundary (Miao et al., 2023, 2024; Zhang et al., 2023). (2) The enhancement extends beyond the relations in R1. As under the w/o R1-Label setting, R1-CF still notably improves the average stable accuracy and reduces the instability ratio. For example, it increases by 4.03% on Qwen-2-7B.(3) In-context learning is an effective way to mitigate

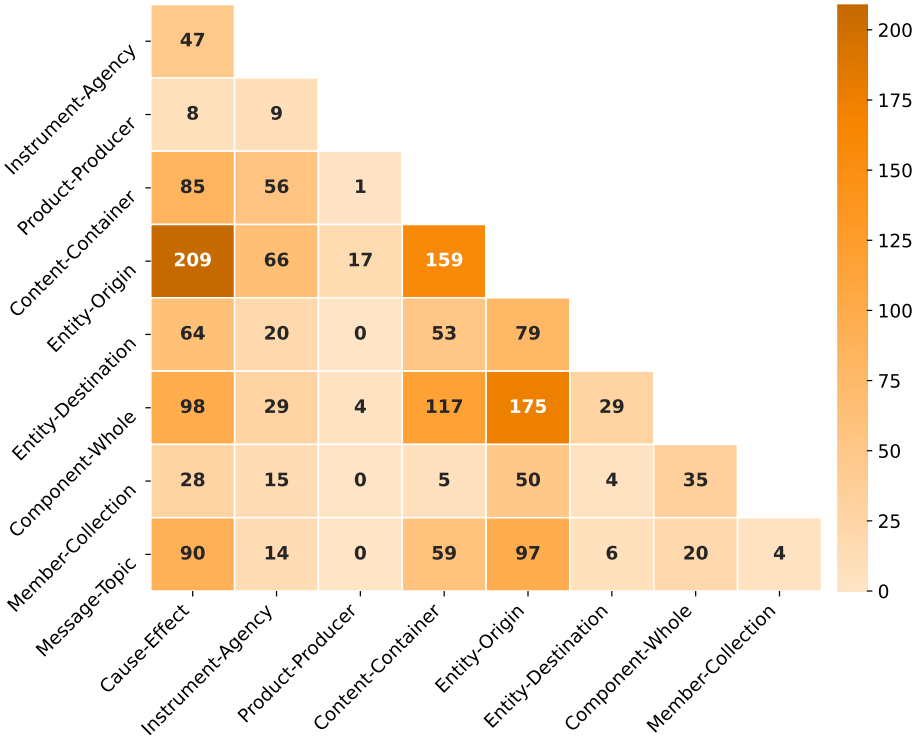

Fig. 12 Error distribution of Llama-3-8B on the test set for RE under SCE evaluation. Darker colors indicate that more instances are misclassified between the specific relation pairs

Table 6 Robustness enhancement results on the validation set for RE (%)

Method	Llama-3-8B				Qwen-2-7B			
	Acc	$\overline{Ins@n}\downarrow$	$\overline{Acc_s[n]}$	$\overline{\Delta Acc_s}$	*Acc*	$\overline{Ins@n}\downarrow$	$\overline{Acc_s[n]}$	$\overline{\Delta Acc_s}$
In-Context (Original)	52.32	55.52	23.27	–	63.93	36.48	40.61	–
w/ CoT	<u>56.19</u>	60.97	21.93	− 1.34	**68.27**	42.33	39.37	− 1.24
w/ R1-Instruction	42.11	62.87	15.63	− 7.64	60.37	35.73	38.80	− 1.81
w/ R1-Example	51.08	<u>49.39</u>	25.85	+ 2.58	<u>67.18</u>	36.25	<u>42.83</u>	+<u>2.22</u>
w/ R1-CF	54.64	**46.74**	**29.10**	+ **5.83**	66.56	34.88	**43.34**	+ **2.73**
w/ R2-CF	**56.50**	53.42	26.32	+ 3.05	63.78	**34.06**	42.05	+ 1.44
w/ R3-CF	54.49	51.14	<u>26.63</u>	+<u>3.36</u>	63.47	<u>34.47</u>	41.59	+ 0.98
w/o R1-Label (Original)	52.32	22.09	40.76	-	63.93	24.29	48.40	–
w/o R1-Label (R1-CF)	54.64	21.15	43.09	+ 2.33	66.56	21.24	52.43	+ 4.03

$\overline{Ins@n}$ and $\overline{Acc_s[n]}$ denote the average instability ratio and stable accuracy, respectively ($n = 1, 2, 3$). $\overline{\Delta Acc_s}$ denotes the improvement in average stable accuracy over the original method. **Bold** and <u>underlined</u> indicate the best and second-best results, respectively. w/: "with". w/o: "without", CF: "Counterfactual"

spurious correlations. As R1-Example improves robustness with standard in-context learning examples and substantially outperforms R1-Instruction.

6.6 Empirical Study

In this section, we provide a detailed explanation of the tolerance rule in Sect. 5.4. The error tolerance is used to adjust the acceptable range of errors, which needs to balance both accuracy and effectiveness. If the tolerance is too strict, fewer biased words will be identified, reducing the effectiveness of the evaluation. Conversely, if it is too lenient, non-biased words may be included, lowering accuracy. In determining the appropriate value, we conduct an empirical study, as shown in Fig. 13. When the value is below 0.3, the number of detected biased words is relatively small, leading to insufficient evaluation effectiveness. When the value exceeds 0.3, the accuracy of detected biased words decreases; even though more words are found, the improvement becomes marginal or even declines. For example, Qwen-2-7B shows a significant performance drop in RE. Thus, we set the error tolerance to a moderate value of 0.3.

6.7 Generalizability Study

In this section, we evaluate the generalizability of our method on the external knowledge base ConceptNet (Speer et al., 2017). The results, shown in Fig. 14, indicate that even when the knowledge base is incomplete or biased, our method consistently maintains relatively stable performance. Specifically, in this experiment, we introduce two types of perturbations: (1) randomly removing 30% of the candidate words to make the knowledge base incomplete, and (2) constructing local bias by replacing 30% of the candidate words with the related words of the remaining 70%. The results show that even under incomplete or biased conditions, models exhibit relatively stable performance in RE. This finding indicates that our framework has a relatively low degree of coupling with the external knowledge base, verifying its strong generalizability. ConceptNet only provides us with an approximate search space, within which our method maintains relatively stable performance based on conditional independence testing.

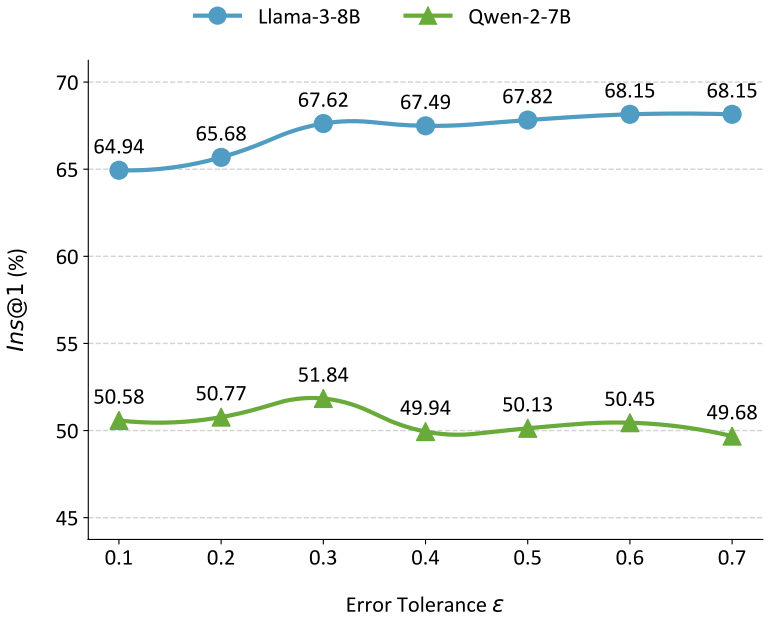

Fig. 13 Empirical study of the error tolerance setting on RE, using *Ins*@1 as the evaluation metric

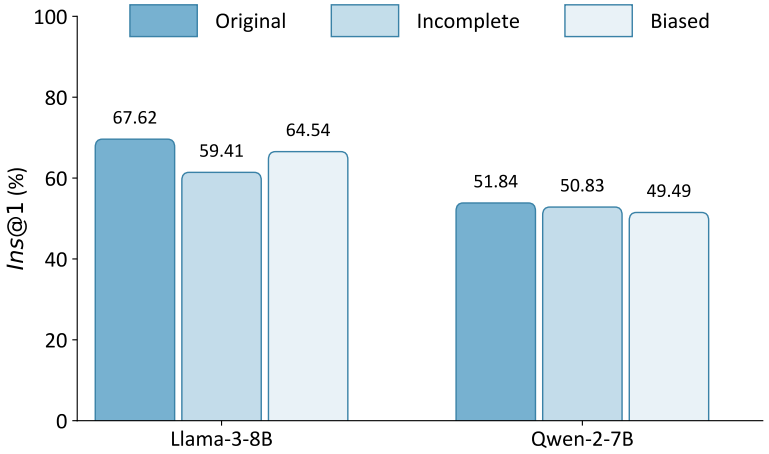

Fig. 14 The generalizability study of ConceptNet in RE. Incomplete denotes the absence of some candidate words, whereas Biased refers to the presence of local bias

6.8 Prompt Sensitivity

In this section, we conduct an analysis of prompt sensitivity to gain insights into the impact of prompt design on probability computation. We find that variations in prompt expression do not affect the conclusions of our study. Specifically, we alter the prompt expressions by replacing certain words with their synonyms while keeping the semantics unchanged. The experimental results, shown in Fig. 15, demonstrate that Llama-3-8B exhibits minimal performance differences across the three tasks compared with the original setting, and the overall trends remain consistent. This indicates that the sensitivity of our prompt design is low, and consistent conclusions can be drawn as long as the semantic meaning is preserved. The underlying reason lies in our minimalist prompt design principle, as discussed in Sect. 4.5.3. This principle avoids introducing excessive concepts into the prompt, thereby reducing uncontrolled interference and mitigating the impact of prompt design on probability computation.

7 Conclusion

In this paper, we explore word-level spurious correlations within LLMs, a key source of instability in real-world IE scenarios. We introduce **SCE** as the first framework to systematically evaluate the robustness of LLMs against these correlations. During noisy data construction, to address the challenge of identifying spurious correlations in the absence of observational data, specifically the training data of LLMs, we propose a novel module **LCD** based on causal discovery, which utilizes statistical information encoded in model parameters to identify spurious correlations. Extensive evaluations on SOTA LLMs reveal their unexpectedly severe fragility in IE tasks and strong reliance on statistical patterns. Compared to recent robustness evaluation methods, SCE outperforms them by a large margin, establishing an SOTA attack system. Additionally, we find that the evaluation results from SCE serve as effective feedback for enhancing model robustness. In summary, our framework

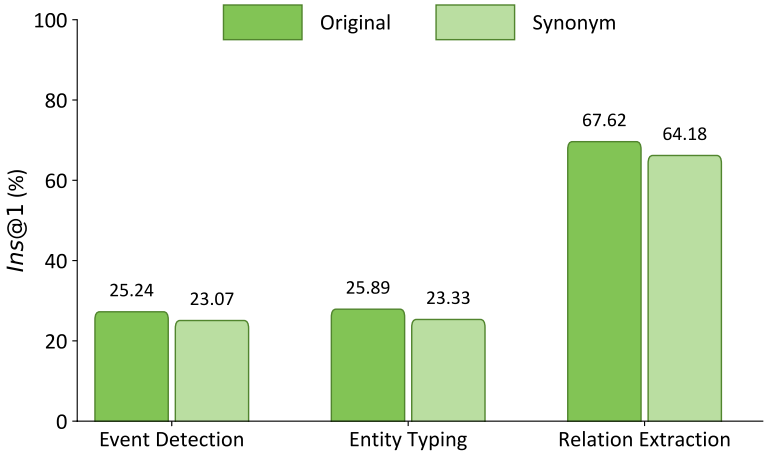

Fig. 15 The analysis of prompt sensitivity on Llama-3-8B in three tasks. Synonym represents the synonym substitution that preserves semantic consistency

quantifies spurious correlations and provides support for identifying and managing potential risks in real-world IE applications. Furthermore, as a strong attack system grounded in causality, it promotes both reliable model development and improved interpretability.

Limitations and future work. Although our proposed SCE framework systematically evaluates semantic spurious correlations of LLMs, it currently focuses only on the IE domain as an initial effort. Future work could extend this framework to a broader range of tasks, leading to a more comprehensive understanding of LLMs. Furthermore, the noisy data are created via rule-based insertions. The effect of spurious correlations may be slightly overestimated, as the insertion rules cannot perfectly preserve semantic consistency. While the semantic integrity of the task is preserved, the resulting sentences may sound unnatural. Future work may focus on developing more sophisticated methods to generate more natural noisy sentences.

Appendix A: Additional Details

A.1 Proof by Contradiction for Joint Probability

This section supplements Sect. 4.5.2. We provide a proof by contradiction showing that LLMs cannot estimate joint probabilities, as LLMs are autoregressive models that generate sequences. While LLMs can estimate conditional probabilities, we apply the chain rule to derive the standard joint probability, denoted as $P(A, B)$:

$$P(A, B) = P(B \mid A)P(A), \tag{A1}$$

where $P(B \mid A)$ and $P(A)$ can be estimated by the LLM-based estimation method in Sect. 4.5.3. Then, we make an assumption that LLMs can directly estimate joint probabilities using prompt: *the most likely to appear simultaneously with A is B*, which is consistent with our probability prompts. We estimate the joint probability through a similar method, except for prompt, which is denoted as $\hat{P}(A, B)$. If most of $\hat{P}(A, B)$ is approximately equal to $P(A, B)$, the assumption is accepted; otherwise, it is rejected.

We randomly select several words for evaluation. The results are shown in Table 7. All differences between the two approaches exceed the error tolerance predefined in Sect. 5.4. Therefore, we reject the assumption and conclude that LLMs are unable to estimate joint probabilities. Accordingly, new conditional independence criteria must be introduced to accommodate the autoregressive nature of LLMs.

A.2 Details of Dataset

This section supplements Sect. 5.1. The words contained in the labels of these datasets are shown in Table 8. In ACE 2005 and SemEval, each label contains multiple words. In

Table 7 Results of the proof by contradiction

Word A	Word B	$P(A, B)$	$\hat{P}(A, B)$	~Equal
Confounder	Causality	4.27e−13	2.82e−12	✗
Causality	Statistics	7.12e−9	2.50e−10	✗
Disease	Injury	2.04e−7	7.33e−9	✗
Injury	Medical	3.12e−8	9.61e−10	✗

Table 8 Labels of the datasets used in the experiments

ACE 2005
Conflict: Attack, Movement: Transport, Contact: Meet, Personnel: End-Position, Transaction: Transfer-money, Life: Injure, Transaction: Transfer-Ownership, Contact: Phone-Write, Personnel: Start-Position, Justice: Trial-Hearing, Justice: Charge-Indict, Justice: Arrest-Jail, Conflict: Demonstrate, Life: Marry, Justice: Convict, Justice: Sue, Life: Die, Personnel: Elect, Justice: Sentence
Few-NERD
Actor, Author, Athlete, Director, Politician, Scholar, Soldier, Airplane, Car, Food, Game, Ship, Software, Weapon
SemEval
Member-Collection, Instrument-Agency, Product-Producer, Message-Topic, Entity-Origin, Entity-Destination, Content-Container, Cause-Effect, Component-Whole

the evaluation process, we identify biased words for each of them, as each word serves as an important clue for the label. To ensure data diversity, we take label type coverage into account when sampling instances from ACE 2005 and Few-NERD. For instance, in Few-NERD, we chose an equal number of labels from the person and object categories. It can be observed that these words are all conceptual terms.

A.3 Details of Prompt

This section supplements Sect. 5.4. We use in-context learning to guide LLMs in performing each task. Table 9 presents the prompt used for relation extraction; those for entity typing and event detection are similar and thus omitted due to space limitations. The prompt primary includes a detailed task instruction and two comprehensive examples. The table presents a pair of counterfactual examples for R1-CF in Sect. 6.5, which flip the label through minimal edits. To standardize the format of model responses, we use special symbols to separate different areas. In addition, to prevent information leakage, examples in the prompt are manually constructed. During evaluation, each instance is inserted into the reserved position in the prompt.

Table 9 In-context learning prompt for relation extraction with two counterfactual examples, which flip the label through minimal edits marked by underlining. {sentence} represents an evaluated sentence

<**Instruction**>
Select the most suitable relation between the given head and tail entities in the given sentence. The relation type must be chosen from the candidate relations.
</**Instruction**>
<**Instance**>
Given Sentence: The smoke comes from the little factory.
Head Entity: smoke
Tail Entity: factory
Candidate Relations: message-topic, entity-origin, entity-destination,
content-container, cause-effect, component-whole, member-collection,
instrument-agency, product-producer
Relation Between the Head Entity and Tail Entity: entity-origin
</**Instance**>
<**Instance**>
Given Sentence: The smoke is caused by the little factory.
Head Entity: smoke
Tail Entity: factory
Candidate Relations: message-topic, entity-origin, entity-destination,
content-container, cause-effect, component-whole, member-collection,
instrument-agency, product-producer
Relation Between the Head Entity and Tail Entity: cause-effect
</**Instance**>
Hint: Complete the remaining content and maintain consistency with the format of the above examples.
<**Instance**>
Given Sentence: {sentence}
Head Entity: {head entity}
Tail Entity: {tail entity}
Candidate Relations: message-topic, entity-origin, entity-destination,
content-container, cause-effect, component-whole, member-collection,
instrument-agency, product-producer
Relation Between the Head Entity and Tail Entity:

{head entity} and {tail entity} denote the given entities

Appendix B: Additional Results

B.1 Main Results

This section supplements Sect. 5.5. Tables 10 and 11 represent the evaluation results of $n = 2, 3$, respectively, which are consistent with the findings summarized in the main text. Furthermore, they reflect that even under more relaxed settings, the instability of LLMs remains a significant issue. For instance, at $n = 3$, the instability ratios on SOTA model

Table 10 Evaluation results (%) of $n = 2$, as a supplement to Sect. 5.5

Model	Event Detection			Entity Typing			Relation Extraction		
	Acc	*Ins*@2 ↓	$Acc_s[2]$	*Acc*	*Ins*@2 ↓	$Acc_s[2]$	*Acc*	*Ins*@2 ↓	$Acc_s[2]$
Llama-3-8B	66.88	20.14	53.41	80.96	17.91	66.46	53.91	55.57	23.95
Llama-3-70B	84.84	10.02	76.34	82.18	8.13	75.50	57.67	18.01	47.28
Llama-3.1-8B	79.71	15.90	67.04	81.25	8.22	74.57	65.89	38.63	40.43
Llama-3.1-70B	84.12	13.73	72.57	**85.71**	12.58	74.93	62.31	16.10	52.28
Llama-3.3-70B	87.73	10.01	78.95	84.25	11.57	74.50	70.22	17.87	57.67
Qwen-1.5-7B	73.06	31.50	50.04	52.75	42.11	30.54	45.16	49.80	22.67
Qwen-1.5-32B	83.88	17.50	69.21	79.82	10.51	71.43	65.89	22.94	50.77
Qwen-1.5-72B	85.85	12.94	74.74	81.61	13.70	70.43	72.43	22.21	56.34
Qwen-2-7B	80.83	22.42	62.71	82.46	16.41	68.93	68.63	41.60	40.08
Qwen-2-72B	88.81	9.21	80.63	84.54	10.27	75.86	74.33	23.31	57.00
Qwen-2.5-7B	79.15	29.74	55.61	81.82	20.82	64.79	67.43	29.49	47.55
Qwen-2.5-32B	87.89	11.54	77.75	84.96	10.55	**76.00**	70.97	24.22	53.78
Qwen-2.5-72B	**90.18**	7.34	**83.56**	84.36	11.60	74.57	77.24	24.03	58.68
Qwen-3-14B	86.57	10.61	77.39	83.25	14.80	70.93	60.10	30.74	41.63
Qwen-3-32B	87.77	15.58	74.10	83.96	15.95	70.57	73.71	24.04	55.99
Qwen-3-32B-Thinking	88.65	31.22	60.97	83.39	22.32	64.78	**79.10**	22.91	**60.98**

Acc: initial accuracy; *Ins*@2: instability ratio; $Acc_s[2]$: stable accuracy. **Bold** and underlined denote the best and second-best results, respectively. Darker indicates greater instability

Table 11 Evaluation results (%) of $n = 3$, as a supplement to Sect. 5.5

Model	Event Detection			Entity Typing			Relation Extraction		
	Acc	*Ins*@3 ↓	$Acc_s[3]$	*Acc*	*Ins*@3 ↓	$Acc_s[3]$	*Acc*	*Ins*@3 ↓	$Acc_s[3]$
Llama-3-8B	66.88	16.13	56.09	80.96	13.32	70.18	53.91	47.62	28.24
Llama-3-70B	84.84	8.88	77.31	82.18	6.35	76.96	57.67	15.56	48.70
Llama-3.1-8B	79.71	14.39	68.24	81.25	6.73	75.79	65.89	30.65	45.69
Llama-3.1-70B	84.12	11.68	74.30	**85.71**	9.58	**77.50**	62.31	12.77	54.35
Llama-3.3-70B	87.73	8.73	80.07	84.25	9.50	76.25	70.22	15.36	59.43
Qwen-1.5-7B	73.06	26.62	53.61	52.75	36.15	33.68	45.16	45.01	24.83
Qwen-1.5-32B	83.88	14.96	71.33	79.82	8.32	73.18	65.89	19.72	52.89
Qwen-1.5-72B	85.85	10.04	77.23	81.61	10.85	72.75	72.43	18.43	59.08
Qwen-2-7B	80.83	18.50	65.88	82.46	13.90	71.00	68.63	36.51	43.57
Qwen-2-72B	88.81	7.54	82.12	84.54	8.91	77.00	74.33	20.21	59.30
Qwen-2.5-7B	79.15	23.56	60.51	81.82	17.29	67.68	67.43	25.82	50.02
Qwen-2.5-32B	87.89	9.26	79.75	84.96	9.37	77.00	70.97	21.42	55.77
Qwen-2.5-72B	**90.18**	6.45	**84.36**	84.36	9.44	76.39	77.24	19.34	62.31
Qwen-3-14B	86.57	9.36	78.47	83.25	12.96	72.46	60.10	26.18	44.37
Qwen-3-32B	87.77	13.48	75.94	83.96	13.10	72.96	73.71	20.20	58.82
Qwen-3-32B-Thinking	88.65	24.89	66.59	83.39	18.45	68.00	**79.10**	18.99	**64.07**

Acc: initial accuracy; *Ins*@3: instability ratio; $Acc_s[3]$: stable accuracy. **Bold** and underlined denote the best and second-best results, respectively. Darker indicates greater instability

Qwen-3-32B remain as high as 13.48%, 13.10%, and 20.20% across the three tasks. This suggests that LLM-based IE systems are still challenging to apply in real-world scenarios, not even mentioning high-risk domains.

B.2 Comparison Results

This section supplements Sect. 5.6. Tables 12 and 13 provide the Comparison results of $n = 2, 3$, respectively, which are consistent with the conclusions summarized in the main text. Even under the more relaxed conditions, SCE consistently outperforms existing robustness evaluation methods. For example, on the RE task with Llama-3-8B, SCE surpasses the second-best method, NEO-BENCH, by 31.66% and 31.66% at $n = 2, 3$, respectively. SCE still serves as an SOTA attack system under multiple conditions.

B.3 Ablation Results

This section supplements Sect. 5.7. Tables 14 and 15 represent the ablation results of $n = 2, 3$, respectively, which are consistent with the conclusions summarized in the main text. The results provide strong evidence for the effectiveness of the LCD module, particularly its two key techniques: the LLM-based conditional independence test and the LLM-based probability estimation method. For example, on the RE task with Llama-3-8B, LCD outperforms RandomRelatedTo by 28.36% and 23.44% at $n = 2, 3$, respectively. Moreover, RandomRelatedTo still outperforms the other two random sampling methods, indicating that related words are more likely to introduce potential spurious correlations, as they are likely to co-occur frequently in the training data.

Table 12 Comparison results of *Ins*@2 (%), as a supplement to Sect. 5.6

Type	Method	Llama-3-8B				Qwen-2-7B			
		ED	ET	RE	Avg	ED	ET	RE	Avg
Classical	StressTest (Naik et al., 2018)	8.81	7.77	14.91	10.50	6.45	3.94	10.31	6.90
	PWWS (Ren et al., 2019)	3.77	4.13	6.06	4.65	4.81	2.73	4.42	3.99
	TextBugger (Li et al., 2019)	9.65	9.60	16.79	12.01	9.22	5.46	11.01	8.56
	TextFooler (Jin et al., 2020)	9.17	9.24	13.02	10.48	9.57	6.80	10.05	8.81
	CheckList (Ribeiro et al., 2020)	9.23	7.86	18.76	11.95	8.23	4.59	11.78	8.20
Classical-Based	PromptRobust (Zhu et al., 2023)	13.48	<u>12.31</u>	23.42	<u>16.40</u>	12.84	8.80	17.22	12.95
	RUPBench (Wang & Zhao, 2024)	<u>13.90</u>	10.40	<u>24.41</u>	16.24	12.15	6.76	18.25	12.39
	ABFS (Xiao et al., 2025)	7.19	7.64	12.04	8.96	7.93	5.94	8.96	7.61
LLM-Specific	NEO-BENCH (Zheng et al., 2024)	13.60	8.09	23.91	15.20	<u>15.67</u>	<u>10.31</u>	<u>23.69</u>	<u>16.56</u>
	ToxiCloakCN (Xiao et al., 2024)	7.07	5.51	15.40	9.33	8.38	6.24	15.24	9.95
	J&H (Hu et al., 2025)	10.84	10.57	22.19	14.53	12.69	7.84	17.67	12.73
	SCE (Ours)†	**20.14**	**17.91**	**55.57**	**31.21**	**22.42**	**16.41**	**41.60**	**26.81**

Bold and <u>underlined</u> denote the best and second-best perturbation performance, respectively. † indicates a statistically significant difference from the second-best group ($p < 0.05$)

Table 13 Comparison results of *Ins*@3 (%), as a supplement to Sect. 5.6

Type	Method	Llama-3-8B				Qwen-2-7B			
		ED	ET	RE	Avg	ED	ET	RE	Avg
Classical	StressTest (Naik et al., 2018)	6.95	6.22	11.79	8.32	5.06	3.29	8.26	5.54
	PWWS (Ren et al., 2019)	2.88	2.71	3.52	3.04	3.52	1.95	2.50	2.66
	TextBugger (Li et al., 2019)	8.45	7.64	14.91	10.33	7.98	4.55	9.35	7.29
	TextFooler (Jin et al., 2020)	8.39	7.95	11.71	9.35	8.87	6.24	8.90	8.00
	CheckList (Ribeiro et al., 2020)	8.27	6.18	16.38	10.28	7.24	3.73	10.56	7.18
Classical-Based	PromptRobust (Zhu et al., 2023)	12.04	10.48	21.21	14.58	11.45	7.02	14.85	11.11
	RUPBench (Wang & Zhao, 2024)	12.52	9.06	21.95	14.51	10.66	5.76	15.75	10.72
	ABFS (Xiao et al., 2025)	6.71	6.49	10.24	7.81	7.04	5.11	8.07	6.74
LLM-Specific	NEO-BENCH (Zheng et al., 2024)	11.92	6.75	21.46	13.38	13.88	8.93	21.25	14.69
	ToxiCloakCN (Xiao et al., 2024)	6.11	4.75	14.33	8.40	7.78	4.94	13.83	8.85
	J&H (Hu et al., 2025)	9.53	9.11	18.84	12.49	10.66	6.76	14.92	10.78
	SCE (Ours)	**16.13**	**13.32**	**47.62**	**25.69**	**18.50**	**13.90**	**36.51**	**22.97**

Bold and underlined denote the best and second-best perturbation performance, respectively

Table 14 Ablation results of *Ins*@2 (%), as a supplement to Sect. 5.7

Method	Llama-3-8B				Qwen-2-7B			
	ED	ET	RE	Avg	ED	ET	RE	Avg
SCE w/ LCD†	**20.14**	**17.91**	**55.57**	**31.21**	**22.42**	**16.41**	**41.60**	**26.81**
LCD → RandomRelatedTo	13.07	9.26	27.21	16.51	15.62	10.35	27.56	17.84
LCD → RandomConceptNet	11.33	7.50	25.16	14.66	12.65	8.10	25.37	15.37
LCD → RandomEnglishWord	12.17	8.60	24.92	15.23	12.35	7.62	25.63	15.20

Bold / underlined: the best / second-best perturbation performance. †: sig. difference from the second-best group ($p < 0.05$). w/: "with". →: replacing LCD with random sampling in a specified space

Table 15 Ablation results of *Ins*@3 (%), as a supplement to Sect. 5.7

Method	Llama-3-8B				Qwen-2-7B			
	ED	ET	RE	Avg	ED	ET	RE	Avg
SCE w/ LCD	**16.13**	**13.32**	**47.62**	**25.69**	**18.50**	**13.90**	**36.51**	**22.97**
LCD → RandomRelatedTo	11.15	7.63	24.18	14.32	12.60	8.14	23.12	14.62
LCD → RandomConceptNet	10.01	6.48	22.95	13.15	11.36	6.80	22.41	13.52
LCD → RandomEnglishWord	10.61	7.01	21.80	13.14	10.81	6.41	22.99	13.40

Bold and underlined denote the best and second-best perturbation performance, respectively. w/: "with". →: replacing LCD with random sampling in a specified space

Table 16 Results of independence assessment, as a supplement to Sect. 6.2

Dependent words($A \not\perp\!\!\!\perp B$)			Independent words ($A \perp\!\!\!\perp B$)		
Word A	Word B	Prediction	Word A	Word B	Prediction
Confounding	Causal Inference	✓	Confounding	Literature	✓
Confounding	Dependent Variable	✓	Confounding	Writings	✓
Confounding	Spurious Association	✓	Confounding	Novels	✓
Confounding	Statistics	✓	Confounding	Poems	✓
Literature	Writings	✓	Literature	Disease	✓
Literature	Novels	✓	Literature	Medical Conditions	✓
Literature	Poems	✓	Literature	Injury	✓
Disease	Medical Conditions	✓	Disease	Symphony	✓
Disease	Injury	✓	Disease	Musical Composition	✓
Disease	Immune System	✓	Disease	Classical Music	✓
Symphony	Musical Composition	✓	Symphony	Stepper	✓
Symphony	Classical Music	✓	Symphony	Integrated Circuits	✓
Symphony	Orchestra	✓	Symphony	Silicon Wafers	✓
Stepper	Integrated Circuits	✓	Stepper	Comics	✓
Stepper	Silicon Wafers	✓	Stepper	Cartoons	✓
Stepper	Memory Chips	✗	Stepper	Graphic Novels	✓
Comics	Cartoons	✓	Comics	Marathon	✓
Comics	Graphic Novels	✓	Comics	Olympic	✗
Comics	Illustration	✓	Comics	Road Race	✓
Marathon	Olympic	✓	Marathon	Transformer	✗
Marathon	Road Race	✓	Marathon	Passive Component	✓
Marathon	Championships	✓	Marathon	Transmission	✓
Transformer	Passive Component	✓	Transformer	Literature	✓
Transformer	Transmission	✓	Transformer	Novels	✓
Transformer	Electromotive Force	✓	Transformer	Poems	✓
Alignment degree		96%	Alignment degree		92%

$A \not\perp\!\!\!\perp B$ indicates that word A is statistically dependent on word B.And $A \perp\!\!\!\perp B$ denotes statistically independent. ✓ and ✗ denote correct and incorrect prediction results, respectively

B.4 Conditional Independence Assessment

This section supplements Sect. 6.2. Tables 16 and 17 detail the manually annotated assessment set and prediction results. All words are sourced from Wikipedia. To prevent ambiguity, we choose words that are either strongly related or entirely unrelated in meaning. The alignment in conditional independence is the lowest at 84%, while all others are above 90%, suggesting that LLMs may encode denser word dependencies than humans, and thus tend to regard words as conditionally dependent.

Table 17 Results of conditional independence assessment, as a supplement to Sect. 6.2

Conditionally dependent words ($A \not\perp\!\!\!\perp B \mid C$)				Conditionally independent words ($A \perp\!\!\!\perp B \mid C$)			
Word A	Word B	Condition C	Pred.	Word A	Word B	Condition C	Pred.
Confounding	Causal Inference	Causal Study	✓	Confounding	Literature	Causal Study	✓
Confounding	Dependent Variable	Causal Study	✓	Confounding	Writings	Causal Study	✓
Confounding	Spurious Association	Causal Study	✓	Confounding	Novels	Causal Study	✓
Confounding	Statistics	Causal Study	✗	Confounding	Poems	Causal Study	✓
Literature	Writings	Publication	✓	Literature	Disease	Publication	✗
Literature	Novels	Publication	✓	Literature	Medical Conditions	Publication	✓
Literature	Poems	Publication	✓	Literature	Injury	Publication	✓
Disease	Medical Conditions	Medical Study	✓	Disease	Symphony	Medical Study	✓
Disease	Injury	Medical Study	✓	Disease	Musical Composition	Medical Study	✓
Disease	Immune System	Medical Study	✓	Disease	Classical Music	Medical Study	✓
Symphony	Musical Composition	Performance	✓	Symphony	Stepper	Performance	✗
Symphony	Classical Music	Performance	✓	Symphony	Integrated Circuits	Performance	✗
Symphony	Orchestra	Performance	✓	Symphony	Silicon Wafers	Performance	✓
Stepper	Integrated Circuits	Device	✓	Stepper	Comics	Device	✓
Stepper	Silicon Wafers	Device	✓	Stepper	Cartoons	Device	✓
Stepper	Memory Chips	Device	✓	Stepper	Graphic Novels	Device	✓
Comics	Cartoons	Entertainment	✓	Comics	Marathon	Entertainment	✓
Comics	Graphic Novels	Entertainment	✓	Comics	Olympic	Entertainment	✓
Comics	Illustration	Entertainment	✓	Comics	Road Race	Entertainment	✓
Marathon	Olympic	Competition	✓	Marathon	Transformer	Competition	✗
Marathon	Road Race	Competition	✓	Marathon	Passive Component	Competition	✓
Marathon	Championships	Competition	✓	Marathon	Transmission	Competition	✓
Transformer	Passive Component	Machine	✓	Transformer	Literature	Machine	✓
Transformer	Transmission	Machine	✓	Transformer	Novels	Machine	✓
Transformer	Electromotive Force	Machine	✓	Transformer	Poems	Machine	✓
Alignment degree			96%	Alignment degree			84%

$A \not\perp\!\!\!\perp B \mid C$ indicates that word A is statistically dependent on word B in the context of word C. On the contrary, $A \perp\!\!\!\perp B \mid C$ denotes statistically conditional independent. ✓ and ✗ denote correct and incorrect prediction results, respectively

Author Contributions Xin Miao: Writing—original draft, Writing—review & editing, Visualization, Validation, Software, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. Yongqi Li: Writing—review & editing, Validation, Investigation, Data curation. Hankun Kang: Writing—review & editing. Mayi Xu: Writing—review & editing. Jintao Wen: Writing—review & editing. Yuyang Ren: Writing—review & editing. Tieyun Qian: Writing—review & editing, Supervision, Resources, Funding acquisition, Conceptualization.

Funding This work was supported by the grant from the National Natural Science Foundation of China (NSFC) project (No.62276193), the Fundamental Research Funds for the Central Universities, China (Grant No. 2042022dx0001) and the Key Laboratory of Computing Power Network and Information Security, Ministry of Education under Grant No. 2024ZD027.

Data Availability No datasets were generated or analysed during the current study.

Materials Availability This study did not generate any new materials. All resources used are publicly available and cited in the manuscript.

Code Availability The code used in this study is publicly available at: https://github.com/NLPWM-WHU/SCE.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Ethical Approval and Consent to Participate This study did not involve any human participants or animals. Ethical approval and informed consent were therefore not required.

Consent for Publication Not applicable. This study does not contain any individual person's data in any form (including individual details, images, or videos).

References

Abdulaal, A., Montana-Brown, N., He, T., Ijishakin, A., Drobnjak, I., Castro, D. C., Alexander, D. C., et al. (2023). Causal modelling agents: Causal graph discovery through synergising metadata-and data-driven reasoning. In *The Twelfth international conference on learning representations*.

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. (2023). GPT-4 technical report. arXiv preprint. arXiv:2303.08774

Andersson, S. A., Madigan, D., & Perlman, M. D. (1997). A characterization of Markov equivalence classes for acyclic digraphs. *The Annals of Statistics, 25*(2), 505–541.

Ashwani, S., Hegde, K., Mannuru, N. R., Sengar, D. S., Jindal, M., Kathala, K. C. R., Banga, D., Jain, V., & Chadha, A. (2024). Cause and effect: Can large language models truly understand causality? In *Proceedings of the AAAI symposium series* (Vol. 4, pp. 2–9).

Baliunas, L. (2023). *Large language models for reliable information extraction*. Department of Engineering, University of Cambridge.

Carlini, N., Tramer, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, U., & et al. (2021). Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)* (pp. 2633–2650).

Chen, H. (2023). Large knowledge model: Perspectives and challenges. arXiv preprint. arXiv:2312.02706

Cheng, Y., Chang, Y., & Wu, Y. (2025). A survey on data contamination for large language models. arXiv e-prints, 2502.

Cui, P., & Athey, S. (2022). Stable learning establishes some common ground between causal inference and machine learning. *Nature Machine Intelligence, 4*(2), 110–115.

Deng, C., Zhao, Y., Tang, X., Gerstein, M., & Cohan, A. (2024). Investigating data contamination in modern benchmarks for large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human language technologies* (Volume 1: Long Papers) (pp. 8698–8711).

Ding, N., Xu, G., Chen, Y., Wang, X., Han, X., Xie, P., Zheng, H., & Liu, Z. (2021). Few-nerd: A few-shot named entity recognition dataset. In *Proceedings of the 59th annual meeting of the Association for Computational Linguistics and the 11th international joint conference on natural language processing* (Volume 1: Long Papers) (pp. 3198–3213).

Dong, Q., Li, L., Dai, D., Zheng, C., Ma, J., Li, R., Xia, H., Xu, J., Wu, Z., Liu, T., et al. (2022). A survey on in-context learning. arXiv preprint arXiv:2301.00234

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 e-prints. arXiv:2407.217832407

Elazar, Y., Kassner, N., Ravfogel, S., Feder, A., Ravichander, A., Mosbach, M., Belinkov, Y., Schütze, H., & Goldberg, Y. (2022). Measuring causal effects of data statistics on language model'sfactual'predictions. arXiv preprint. arXiv:2207.14251

Gan, Y., Yang, Y., Ma, Z., He, P., Zeng, R., Wang, Y., Li, Q., Zhou, C., Li, S., Wang, T., et al. (2024). Navigating the risks: A survey of security, privacy, and ethics threats in LLM-based agents. arXiv preprint. arXiv:2411.09523

Gao, J., Ding, X., Qin, B., & Liu, T. (2023). Ischatgpt a good causal reasoner? A comprehensive evaluation. In *Findings of the Association for Computational Linguistics: EMNLP 2023* (pp. 11111–11126).

Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., & Wichmann, F. A.(2020). Shortcut learning in deep neural networks. *Nature Machine Intelligence, 2*(11), 665–673.

Goel, A., Gueta, A., Gilon, O., Liu, C., Erell, S., Nguyen, L.H., Hao, X., Jaber, B., Reddy, S., Kartha, R., et al. (2023). LLMS accelerate annotation for medical information extraction. In *Machine learning for health (ML4H)* (pp. 82–100). PMLR.

Grishman, R., Westbrook, D., & Meyers, A.(2005). Nyu's English ACE 2005 system description. In *ACE 2005 evaluation workshop*.

Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al. (2025). Deepseek-r1: Incentivizing reasoning capability in LLMS via reinforcement learning. arXiv preprint. arXiv:2501.12948

Hammoudeh, Z., & Lowd, D. (2024). Training data influence analysis and estimation: A survey. *Machine Learning, 113*(5), 2351–2403.

Hendrickx, I., Kim, S. N., Kozareva, Z., Nakov, P., Séaghdha, D. O., Padó, S., Pennacchiotti, M., Romano, L., & Szpakowicz, S. (2010). Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. *ACL, 2010*, 33.

Hobbhahn, M., Lieberum, T., & Seiler, D. (2022). Investigating causal understanding in LLMS. In *NeurIPS ML safety workshop*.

Hu, X., Chen, J., Li, X., Guo, Y., Wen, L., Philip, S.Y., & Guo, Z. (2024). Towards understanding factual knowledge of large language models. In *ICLR*.

Hu, Y., Liu, H., Chen, Q., Zheng, N., Wang, C., Liu, Y., Clarke, C.L., & Shen, W. (2025). J&H: Evaluating the robustness of large language models under knowledge-injection attacks in legal domain. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 39, pp. 28106–28115).

Huang, F., Huang, Q., Zhao, Y., Qi, Z., Wang, B., Huang, Y., & Li, S. (2023). A three-stage framework for event-event relation extraction with large language model. In *International conference on neural information processing* (pp. 434–446). Springer.

Huang, A.H., Wang, H., & Yang, Y. (2023). Finbert: A large language model for extracting information from financial text. *Contemporary Accounting Research, 40*(2), 806–841.

Hui, B., Yang, J., Cui, Z., Yang, J., Liu, D., Zhang, L., Liu, T., Zhang, J., Yu, B., Lu, K., et al. (2024). Qwen2. 5-coder technical report. arXiv preprint. arXiv:2409.12186

Hurst, A., Lerer, A., Goucher, A.P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al. (2024). Gpt-4o system card. arXiv preprint. arXiv:2410.21276

Jin, D., Jin, Z., Zhou, J. T., & Szolovits, P. (2020). Is Bert really robust? A strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, pp. 8018–8025).

Jin, Z., Chen, Y., Leeb, F., Gresele, L., Kamal, O., Lyu, Z., Blin, K., Gonzalez Adauto, F., Kleiman-Weiner, M., Sachan, M., et al. (2024). Cladder: A benchmark to assess causal reasoning capabilities of language models. In *Advances in neural information processing systems* (Vol. 36).

Jin, Z., Liu, J., Zhiheng, L., Poff, S., Sachan, M., Mihalcea, R., Diab, M. T., & Schölkopf, B. (2024). Can large language models infer causation from correlation? In *The twelfth international conference on learning representations*.

Jiralerspong, T., Chen, X., More, Y., Shah, V., Bengio, Y. (2024). Efficient causal graph discovery using large language models. In *ICLR 2024 workshop: How far are we from AGI*.

Kang, C., & Choi, J. (2023). Impact of co-occurrence on factualknowledge of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023* (pp. 7721–7735).

Kiciman, E., Ness, R., Sharma, A., Tan, C.: Causal reasoning and large language models: Opening a new frontier for causality. Transactions on machine learning research (2024)

Kim, Y., Guo, L., Yu, B., & Li, Y. (2023). Can ChatGPT understand causal language in science claims? In *13th Workshop on computational approaches to subjectivity, sentiment and social media analysis, WASSA 2023* (pp. 379–389). Association for Computational Linguistics (ACL).

Li, H., Gao, H., Wu, C., & Vasarhelyi, M. A.(2023). Extracting financial data from unstructured sources: Leveraging large language models. *Journal of Information Systems*. https://doi.org/10.2139/ssrn.4567607

Li, J., Ji, S., Du, T., Li, B., & Wang, T. (2019). Textbugger: Generating adversarial text against real-world applications. In *26th Annual network and distributed system security symposium, NDSS 2019*. The Internet Society.

Li, Y., Guo, Y., Guerin, F., & Lin, C. (2024). An open-source data contamination report for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024* (pp. 528–541)

Liu, C., Chen, Y., Liu, T., Gong, M., Cheng, J., Han, B., & Zhang, K. (2024). Discovery of the hidden world with large language models. arXiv preprint. arXiv:2402.03941

Liu, X., Xu, P., Wu, J., Yuan, J., Yang, Y., Zhou, Y., Liu, F., Guan, T., Wang, H., Yu, T., McAuley, J., Ai, W., & Huang, F. (2025). Large language models and causal inference in collaboration: A comprehensive survey. In *Findings of the Association for Computational Linguistics: NAACL 2025* (pp. 7668–7684).

Long, S., Piché, A., Zantedeschi, V., Schuster, T., & Drouin, A. (2023). Causal discovery with language models as imperfect experts. arXiv preprint. arXiv:2307.02390

Long, S., Schuster, T., & Piché, A. (2022). Can large language models build causal graphs? In *NeurIPS 2022 workshop on causal machine learning for real-world impact (CML4Impact 2022)*.

Lu, Y., Liu, Q., Dai, D., Xiao, X., Lin, H., Han, X., Sun, L., & Wu, H. (2022). Unified structure generation for universal information extraction. In *Proceedings of the 60th annual meeting of the Association for Computational Linguistics* (Volume 1: Long Papers) (pp. 5755–5772).

Miao, X., Li, Y., & Qian, T. (2023). Generating commonsense counterfactuals for stable relation extraction. In *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 5654–5668).

Miao, X., Li, Y., Zhou, S., & Qian, T. (2024). Episodic memory retrieval from LLMS: A neuromorphic mechanism to generate commonsense counterfactuals for relation extraction. In *Findings of the Association for Computational Linguistics ACL 2024* (pp. 2489–2511).

Monajatipoor, M., Yang, J., Stremmel, J., Emami, M., Mohaghegh, F., Rouhsedaghat, M., & Chang, K.-W. (2024). LLMS in biomedicine: A study on clinical named entity recognition. arXiv preprint. arXiv:2404.07376

Mondal, I., & Sancheti, A. (2024). How much reliable is chatgpt's prediction on information extraction under input perturbations? arXiv preprint. arXiv:2404.05088

Naik, A., Ravichander, A., Sadeh, N., Rose, C., & Neubig, G. (2018). Stress test evaluation for natural language inference. In *Proceedings of the 27th international conference on computational linguistics* (pp. 2340–2353).

Pearl, J. (2000). *Causality: Models, reasoning, and inference*. Cambridge University Press.

Pearl, J. (2009). *Causality: Models, reasoning, and inference*. Cambridge University Press.

Pearl, J., & Mackenzie, D. (2018). *The book of why: the new science of cause and effect*. Basic Books.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI Blog, 1*(8), 9.

Rajpoot, P., & Parikh, A. (2023). Gpt-finre: In-context learning for financial relation extraction using large language models. In *Proceedings of the sixth workshop on financial technology and natural language processing* (pp. 42–45).

Raza, S., Bamgbose, O., Ghuge, S., Tavakoli, F., Reji, D. J., & Bashir, S. R. (2025). Developing safe and responsible large language model: can we balance bias reduction and language understanding? *Machine Learning, 114*(6), 140.

Reimers, N., & Gurevych, I. (2019). Sentence-Bert: Sentence embeddings using Siamese Bert-networks. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)* (pp. 3982–3992).

Ren, S., Deng, Y., He, K., & Che, W. (2019). Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th annual meeting of the Association for Computational Linguistics* (pp. 1085–1097).

Ribeiro, M.T., Wu, T., Guestrin, C., & Singh, S. (2020). Beyond accuracy: Behavioral testing of nlp models with checklist. In *Proceedings of the 58th annual meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Romanou, A., Montariol, S., Paul, D., Laugier, L., Aberer, K., & Bosselut, A. (2023). Crab: Assessing the strength of causal relationships between real-world events. In *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 15198–15216).

Sakib, M. N., Islam, M. A., Pathak, R., & Arifin, M. M. (2024). Risks, causes, and mitigations of widespread deployments of large language models (LLMS): A survey. In *2024 2nd International conference on artificial intelligence, blockchain, and Internet of Things (AIBThings)* (pp. 1–7). IEEE.

Sarawagi, S., et al. (2008). Information extraction. *Foundations and Trends® in Databases*, *1*(3), 261–377.

Shahriar, S., Lund, B.D., Mannuru, N. R., Arshad, M. A., Hayawi, K., Bevara, R. V. K., Mannuru, A., & Batool, L. (2024). Putting GPT-4O to the sword: A comprehensive evaluation of language, vision, speech, and multimodal proficiency. *Applied Sciences, 14*(17), 7782.

Simon, H. A.(1954). Spurious correlation: A causal interpretation. *Journal of the American Statistical Association, 49*(267), 467–479.

Singh, A., Singh, N., & Vatsal, S. (2024). Robustness of llms to perturbations in text. arXiv preprint. arXiv:2407.08989

Speer, R., Chin, J., & Havasi, C. (2017). Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 31).

Spirtes, P., & Zhang, K. (2016). Causal discovery and inference: Concepts and recent methodological advances. *Applied Informatics*, *3*, 1–28.

Spirtes, P., Glymour, C. N., & Scheines, R. (2000). *Causation, prediction, and search*. MIT.

Sun, K., Zhang, R., Mensah, S., Mao, Y., & Liu, X. (2020). Recurrent interaction network for jointly extracting entities and classifying relations. arXiv preprint. arXiv:2005.00162

Tiwari, K., Yuan, S., & Zhang, L. (2022). Robust hate speech detection via mitigating spurious correlations. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th international joint conference on natural language processing* (Volume 2: Short Papers) (pp. 51–56).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (Vol. 30).

Verma, T., & Pearl, J. Equivalence and synthesis of causal models. In: Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence, pp. 255–270 (1990).

Villalobos, P., Ho, A., Sevilla, J., Besiroglu, T., Heim, L., & Hobbhahn, M. (2024). Position: Will we run out of data? Limits of LLM scaling based on human-generated data. In *International conference on machine learning* (pp. 49523–49544). PMLR.

Wan, Z., Cheng, F., Mao, Z., Liu, Q., Song, H., Li, J., & Kurohashi, S. (2023). GPT-RE: In-context learning for relation extraction using large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 3534–3547).

Wan, G., Wu, Y., Hu, M., Chu, Z., & Li, S. (2024). Bridging causal discovery and large language models: A comprehensive survey of integrative approaches and future directions. arXiv preprint. arXiv:2402.11068

Wang, S., Sun, X., Li, X., Ouyang, R., Wu, F., Zhang, T., Li, J., & Wang, G. (2023). GPT-NER: Named entity recognition via large language models. arXiv preprint. arXiv:2304.10428

Wang, Q., Ding, K., Liang, B., Yang, M., & Xu, R. (2023). Reducing spurious correlations in aspect-based sentiment analysis with explanation from large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023* (pp. 2930–2941)

Wang, Y., Chen, M., Zhou, W., Cai, Y., Liang, Y., Liu, D., Yang, B., Liu, J., & Hooi, B. (2022). Should we rely on entity mentions for relation extraction? debiasing relation extraction with counterfactual analysis. In *Proceedings of the 2022 conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 3071–3081).

Wang, Y., & Zhao, Y. (2024). RUPBENCH: Benchmarking reasoning under perturbations for robustness evaluation in large language models. arXiv preprint. arXiv:2406.11020

Willig, M., Zečević, M., Dhami, D. S., & Kersting, K. (2023). Probing for correlations of causal facts: Large language models and causality. In *The 11th International conference on learning representations (ICLR 2023)*.

Wu, A., Kuang, K., Zhu, M., Wang, Y., Zheng, Y., Han, K., Li, B., Chen, G., Wu, F., & Zhang, K. (2024). Causality for large language models. CoRR. arXiv:2410.15319 [cs.CL]

Wu, S., Li, D., Ye, H., Chen, Z., Zhou, J., Lou, J., Zheng, Z., & Ng, S.-K. (2025). Tsrating: Rating quality of diverse time series data by meta-learning from LLM judgment. arXiv preprint. arXiv:2506.01290

Wu, J., Yu, T., Chen, X., Wang, H., Rossi, R., Kim, S., Rao, A., & McAuley, J. (2024) DECOT: Debiasing chain-of-thought for knowledge-intensive tasks in large language models via causal intervention. In *Proceedings of the 62nd annual meeting of the Association for Computational Linguistics* (Volume 1: Long Papers) (pp. 14073–14087).

Xiao, M., Xiao, Y., Ji, S., Li, Y., Xue, L., & Zhang, P. (2025). ABFS: Natural robustness testing for llm-based nlp software. arXiv preprint. arXiv:2503.01319

Xiao, Y., Hu, Y., Choo, K., & Lee, R. (2024). TOXICLOAKCN: Evaluating robustness of offensive language detection in chinese with cloaking perturbations. In *Proceedings of the 2024 conference on empirical methods in natural language processing* (pp. 6012–6025).

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. (2025). QWEN3 technical report. arXiv preprint. arXiv:2505.09388

Ye, W., Zheng, G., Cao, X., Ma, Y., & Zhang, A. (2024). Spurious correlations in machine learning: A survey. arXiv preprint. arXiv:2402.12715

Yu, T., Jing, Y., Zhang, X., Jiang, W., Wu, W., Wang, Y., Hu, W., Du, B., & Tao, D. (2025). Benchmarking reasoning robustness in large language models. arXiv preprint. arXiv:2503.04550

Yu, T., Yang, M., Li, C., & Xu, R. (2023). Reducing spurious correlations for relation extraction by feature decomposition and semantic augmentation. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval* (pp. 2324–2328).

Yuan, J., Zheng, H., Yu, H., & Luo, X. (2025). Entangle-then-disentangle: A novel approach for enhancing large vision-language model. *Machine Learning, 114*(8), 1–28.

Zecevic, M., Willig, M., Dhami, D. S., & Kersting, K. (2023). Causal parrots: Large language models may talk causality but are not causal. *Transactions on Machine Learning Research, 2023*(8), 1–27.

Zhang, C., Zhang, L., Wu, J., He, Y., & Zhou, D. (2025). Causal prompting: Debiasing large language model prompting based on front-door adjustment. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 39, pp. 25842–25850).

Zhang, M., Qian, T., Zhang, T., & Miao, X. (2023). Towards model robustness: Generating contextual counterfactuals for entities in relation extraction. In *Proceedings of the ACM web conference 2023* (pp. 1832–1842).

Zhang, W., Lu, W., Wang, J., Wang, Y., Chen, L., Jiang, H., Liu, J., & Ruan, T. (2024). Unexpected phenomenon: LLMS' spurious associations in information extraction. In *Findings of the Association for Computational Linguistics ACL 2024* (pp. 9176–9190).

Zhao, B., Zhang, Y., Xu, Z., Ren, Y., Zhang, X., Luo, R., Feng, Z., & Xia, F. (2025). Unbiased reasoning for knowledge-intensive tasks in large language models via conditional front-door adjustment. arXiv preprint arXiv:2508.16910

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., et al. (2023). A survey of large language models. arXiv e-prints, 2303.

Zheng, J., Ritter, A., & Xu, W. (2024). Neo-bench: Evaluating robustness of large language models with neologisms. In *Proceedings of the 62nd annual meeting of the Association for Computational Linguistics* (Volume 1: Long Papers), (pp. 13885–13906).

Zhou, Y., Xu, P., Liu, X., An, B., Ai, W., & Huang, F. (2024). Explore spurious correlations at the concept level in language models for text classification. In *Proceedings of the 62nd annual meeting of the Association for Computational Linguistics* (Volume 1: Long Papers) (pp. 478–492).

Zhu, K., Wang, J., Zhou, J., Wang, Z., Chen, H., Wang, Y., Yang, L., Ye, W., Zhang, Y., Gong, N., et al. (2023). Promptrobust: Towards evaluating the robustness of large language models on adversarial prompts. In *Proceedings of the 1st ACM workshop on large AI systems and models with privacy and safety analysis* (pp. 57–68).