Project Documentation
Robustness in Counterfactuals Generation using LLMs

Chang Sun, Xiaofeng Zhang
Course Code: 91257-Ethics in Artificial Intelligence

June 4, 2025

Abstract

In explainable artificial intelligence (XAI), counterfactual explanations (CEs) act as an efficient post-hoc method to provide actionable insights by identifying the minimal changes to input features that would alter a model's prediction. Large language models (LLMs) have emerged as powerful tools for CE generation due to their strong capabilities and common sense reasoning. However, current evaluation frameworks suffer from a critical flaw: the unreliable assumption that Oracle models provide perfect ground truth for counterfactual assessment.

Our analysis of the Oracle model (michelecafagna26/t5-base-finetuned-sst2-sentiment) reveals significant reliability issues: 45% error rate (9/20 false negatives) and consistently low confidence scores (0.5-0.6 range). This Oracle unreliability fundamentally undermines traditional Label Flip Rate (LFR) evaluation, which cannot distinguish between harmful manipulation of correct Oracle decisions and beneficial correction of Oracle errors. We propose a disaggregated evaluation framework that separates True Negative (TN) and False Negative (FN) flip rates, enabling proper assessment of LLM behavior when Oracle reliability is questionable. This work demonstrates the urgent need for Oracle-aware evaluation frameworks in real-world XAI applications.

1 Introduction

Explainable AI (XAI) has become an essential component of research, development, and increasingly, the development of Machine Learning (ML) systems ([2, 1, 3]). The motivation for this is twofold. First, it is crucial for data science practitioners and end-users to understand how ML models operate. Second, policy regulations such as the EU's General Data Protection Regulation (GDPR) mandate that automated decision-making–under which many applied ML systems fall–must be explainable to individuals.

The appeal of counterfactuals (CFs) lies in their alignment with the intuitive, "what-if" reasoning that humans commonly use to understand events in both science and everyday life. Rooted in long-standing traditions in philosophy and artificial intelligence [6], CFs have more recently gained traction as a practical tool for explaining ML outcomes–often without a strict commitment to causal interpretation.

When evaluating the CFs generated by LLMs, most of the works consider only Flip

Rate, Token Distance and Probability Change [4, 5], which are measured by the Oracle. Traditionally we treat the Oracle as the ground truth without thinking that it can make mistakes too. To lead to a fairer AI, we propose two metrics for CFs evaluation:

- **Label Flip Rate**: We feed the CFs generated by LLMs back to the Oracle to see how frequently they can flip the original label. What's more, contrary to tradition work, we only take TN and FN samples from the Oracle for CFs generation. Under the assumption that the Oracle is the ground truth, we only need to focus on how to flip the label with the Oracle, not in reality.
- **Self-correction Rate**: Given that the Oracle can be wrong, this metric is designed to evaluate LLM's ability to recognize and correct an oracle's mistake. We take misclassified samples (FN and FP) from the Oracle and let LLMs to generate CFs without telling them which sentiment to flip to. After this we feed the generated CFs back to LLMs for classification. If the LLM can generate a CF that is opposite to the true label of the original sample, then we think this LLM successfully recognized the error made by the Oracle.

2 Methodology

2.1 Task Formulation

Given a dataset $\mathcal{D} = \{x^i; y^i\}^N$ of a classification task where x represents each input sentence and $y \in \{0, 1\}$, we have an Oracle $\mathcal{O}$ which we deem as the source of uncertainty. For each x^i in the dataset, the predictions from $\mathcal{O}$ are $\mathcal{O}_{pred} \in \{TN, TP, FN, FP\}$ where TN, TP, FN, FP means TrueNegative, TruePositive, FalseNegative, FalsePositive individually. We then use a LLM M to generate counterfactuals of each x^i, denoted as CF^i. To measure the quality of generated counterfactuals under the assumption that the Oracle can be wrong, we design two metrics: 1) Label Flip Rate(LFR) and 2) Self-correction Rate(SCR).

2.2 Label Flip Rate Analysis

Counterfactuals, where minimal changes to inputs flip a model's prediction, are particularly powerful for banks and stakeholders. For instance, consider a scenario where an individual's loan application is rejected by a bank; a counterfactual explanation would clearly outline the smallest adjustments the applicant could make to their profile to obtain approval. In this case, an Oracle is the machine learning (ML) tool bank uses for application check.

However, traditional Label Flip Rate (LFR) analysis suffers from a fundamental flaw: it assumes Oracle reliability. Our analysis reveals that the Oracle model exhibits significant unreliability, making traditional LFR evaluation inadequate. We must distinguish between **(1) manipulation of correct Oracle decisions** and **(2) correction of Oracle errors**, as these represent fundamentally different scenarios that require separate evaluation.

Disaggregated LFR Framework Given the LLM M which generates a counterfactual sentence, and the Oracle $\mathcal{O}$ which is a fixed classifier:

$$\mathcal{F}^i = M(x^i) \tag{1}$$

$$\mathcal{O} : x \mapsto \hat{y}_{\mathcal{O}}(x) \in \{0, 1\} \tag{2}$$

We separate Oracle-negative samples into two ethically distinct categories:

$$S_{TN} = \{i : \mathcal{O}^i_{\text{pred}} = TN\} \quad \text{(Oracle correctly predicts negative)} \tag{3}$$

$$S_{FN} = \{i : \mathcal{O}^i_{\text{pred}} = FN\} \quad \text{(Oracle incorrectly rejects positive)} \tag{4}$$

LFR Definitions For True Negative samples, flipping represents potential **harmful manipulation**:

$$\text{TN Flip Rate} = \frac{1}{|S_{TN}|} \sum_{i \in S_{TN}} \mathbf{1}[\hat{y}_{\mathcal{O}}(\mathcal{F}^i) = 1] \tag{5}$$

For False Negative samples, flipping represents potential **beneficial bias correction**:

$$\text{FN Flip Rate} = \frac{1}{|S_{FN}|} \sum_{i \in S_{FN}} \mathbf{1}[\hat{y}_{\mathcal{O}}(\mathcal{F}^i) = 1] \tag{6}$$

We introduce a **Moral Score** to capture the ethical balance:

$$\text{Moral Score} = \text{FN Flip Rate} - \text{TN Flip Rate} \tag{7}$$

where positive scores indicate more bias correction than manipulation, while negative scores suggest the opposite.

Traditional LFR as Aggregate The traditional LFR becomes:

$$\text{Total LFR} = \frac{|S_{TN}| \cdot \text{TN Flip Rate} + |S_{FN}| \cdot \text{FN Flip Rate}}{|S_{TN}| + |S_{FN}|} \tag{8}$$

2.3 Self-correction Rate

Given the assumption that the Oracle might be wrong, Self-correction Rate (SCR) aims to evaluate LLM's ability to recognize and correct an oracle's mistake. For each Oracle error - each FP or FN case - we perform a two-step with the LLM:

1. **CF Generation**: We give LLM the original misclassified input without revealing the true label. The LLM must decide on its own which sentiment to flip to.
2. **LLM Classification**: Next we ask the LLM to classify the sentiment of the counterfactual it just wrote.

In simpler terms, Self-correction is confirmed when:

- For an FN case, the LLM produces a negative sentiment counterfactual and labels it negative. This suggests that the LLM understood the original should be positive.
- For an FP case, the LLM should produce a positive sentiment counterfactual and labels it positive.

Self-correction Rate definition Formally, let L be all indices of misclassified samples (FP $\cup$ FN):

$$L = \{i : \mathcal{O}^i_{\text{pred}} \in \{FP, FN\}\} \tag{9}$$

For each such sample $k \in L$, let $\hat{y}_{\mathcal{O}}(x^k)$ be the Oracle's incorrect prediction for the original text x^k, and let $\hat{y}_{\text{LLM}}(\mathcal{F}^k)$ be the label that the LLM assigns to the counterfactual it generated. Then we define:

$$\text{SCR} = \frac{1}{|L|} \sum_{k \in L} \mathbf{1}\{\hat{\mathbf{y}}_{\mathbf{LLM}}(\mathcal{F}^{\mathbf{k}}) = \hat{\mathbf{y}}_{\mathcal{O}}(\mathbf{x}^{\mathbf{k}})\} \tag{10}$$

where $\mathbf{1}\{\cdot\}$ denotes the indicator function and $|\cdot|$ is the

3 Experiments

3.1 Experimental Setup

3.1.1 Oracle-based Counterfactual Generation

We conducted experiments using a sentiment classification task on the SST-2 dataset, employing a T5-based model fine-tuned for sentiment analysis as our Oracle classifier. This Oracle represents a typical black-box model that stakeholders might encounter in real-world applications. We processed samples from the SST-2 validation set and identified Oracle-negative instances for LFR evaluation.

Dataset and Oracle Performance The Oracle demonstrated specific classification patterns in our experiments. When processing samples from the SST-2 validation set, we focused on Oracle-negative predictions (samples where $\hat{y}_{\mathcal{O}}(x^i) = 0$) as defined in our theoretical framework. These Oracle-negative samples, comprising both True Negatives (TN) and False Negatives (FN), form the foundation for our LFR calculation.

Models We evaluated four 7B-parameter LLMs for counterfactual generation:

- Falcon-7B: A causal decoder-only model trained on a mixture of web data, known for its strong performance on general knowledge tasks but with less instruction-following optimization
- Mistral-7B-Instruct: An instruction-tuned model with strong reasoning capabilities and enhanced context handling, designed to follow complex instructions
- Vicuna-7B: A fine-tuned LLaMA model optimized for dialogue and conversational tasks, with improved instruction following
- Zephyr-7B: A model fine-tuned with direct preference optimization, specifically designed to align with human preferences and follow instructions accurately
- Qwen2-7B-Instruct: An instruction-tuned version of Qwen large language models.

Prompt Design We employed a Chain-of-Thought (CoT) prompting strategy to guide the models in generating counterfactuals. For LFR evaluation, all Oracle-negative samples target positive sentiment to flip the Oracle's prediction from 0 to 1:

```
We want to generate a counterfactual for the following instance.

Original text: "{text}"
Target label: positive

Please follow these three steps:
• Step 1: Identify all of the important words that contribute to the
↪ negative sentiment.
• Step 2: Find replacements for the words identified in Step 1 that lead
↪ to positive sentiment.
• Step 3: Replace the words from Step 2 in the original text to obtain
↪ the counterfactual instance.

Output only the final counterfactual sentence that expresses positive
↪ sentiment.
```

Each model received this prompt template adapted to its specific input format requirements. The consistent target of positive sentiment ensures proper LFR evaluation according to our mathematical definition.

Generation Parameters We tailored generation parameters based on the target sentiment direction to optimize counterfactual quality:

- For positive→negative transformations: Lower temperature (0.3-0.4), higher beam search width (5-8), and stricter constraints to ensure sentiment flipping. These conservative parameters help maintain semantic similarity while ensuring sufficient negative sentiment.
- For negative→positive transformations: Higher temperature (0.5-0.7), moderate beam search width (3-5), and relaxed constraints, allowing more creative generations. The higher temperature encourages more diverse word choices to overcome the Oracle's positive bias threshold.

We also implemented additional generation controls including repetition penalties (1.2-1.5) to prevent the models from repeating phrases, and top-p sampling (0.92-0.95) to maintain output coherence while allowing for creativity.

3.2 Results and Analysis

3.2.1 Label Flip Rate Analysis

Our LFR analysis reveals critical ethical distinctions hidden within traditional aggregate metrics. We evaluate both the Total LFR and the disaggregated TN/FN flip rates, introducing the Moral Score to capture the ethical complexity of counterfactual generation.

Disaggregated Performance Results Table 1 presents our comprehensive analysis across all tested models.

Model	TN Rate	FN Rate	Total	Score	Assessment
Zephyr-7B	0% (0/11)	0% (0/9)	0%	0.0	**Complete Restraint**
Falcon-7B	90.9% (10/11)	88.9% (8/9)	90%	-2.0	High Performance
Mistral-7B	100% (11/11)	100% (9/9)	100%	0.0	Perfect Performance
Vicuna-7B	100% (11/11)	100% (9/9)	100%	0.0	Perfect Performance

Table 1: LFR Analysis: TN Rate (potential manipulation), FN Rate (potential bias correction), Total LFR, Moral Score, and Ethical Assessment for each model.

Oracle Reliability Issues Revealed Our analysis exposes critical Oracle reliability problems that traditional LFR metrics fail to address:

Oracle Performance Analysis:

- **High Error Rate**: 45% false negative rate (9/20 samples) - Oracle incorrectly rejects positive sentiment
- **Low Confidence**: Most decisions have 0.5-0.6 confidence scores, indicating Oracle uncertainty
- **Specific Errors**: Oracle misclassifies clearly positive sentences like "it's a charming and often affecting journey"

Impact on LFR Evaluation:

- **Zephyr-7B (0% flip rate)**: Complete avoidance of Oracle interaction - avoids both manipulation and correction
- **High-performing models (90-100% flip rate)**: May be correcting Oracle errors rather than manipulating correct decisions
- **Traditional Interpretation**: Misleadingly treats all flips as manipulation when many are error correction

This reveals the fundamental problem: *without considering Oracle reliability, LFR evaluation cannot distinguish between beneficial error correction and harmful manipulation.*

LLM Behavior Analysis

- **Mistral-7B & Vicuna-7B**: Perfect performance with 100% flip rates across both categories. These models demonstrate high capability for generating effective counterfactuals regardless of Oracle correctness.
- **Falcon-7B**: Strong performance with 90.9% TN flip rate and 88.9% FN flip rate. The slight difference (Moral Score -2.0) indicates minor variation in success rates between Oracle-correct and Oracle-error scenarios.
- **Zephyr-7B**: Complete restraint (0% both categories) suggests either strong safety constraints preventing Oracle manipulation or fundamental limitations in counterfactual generation capability for this domain.

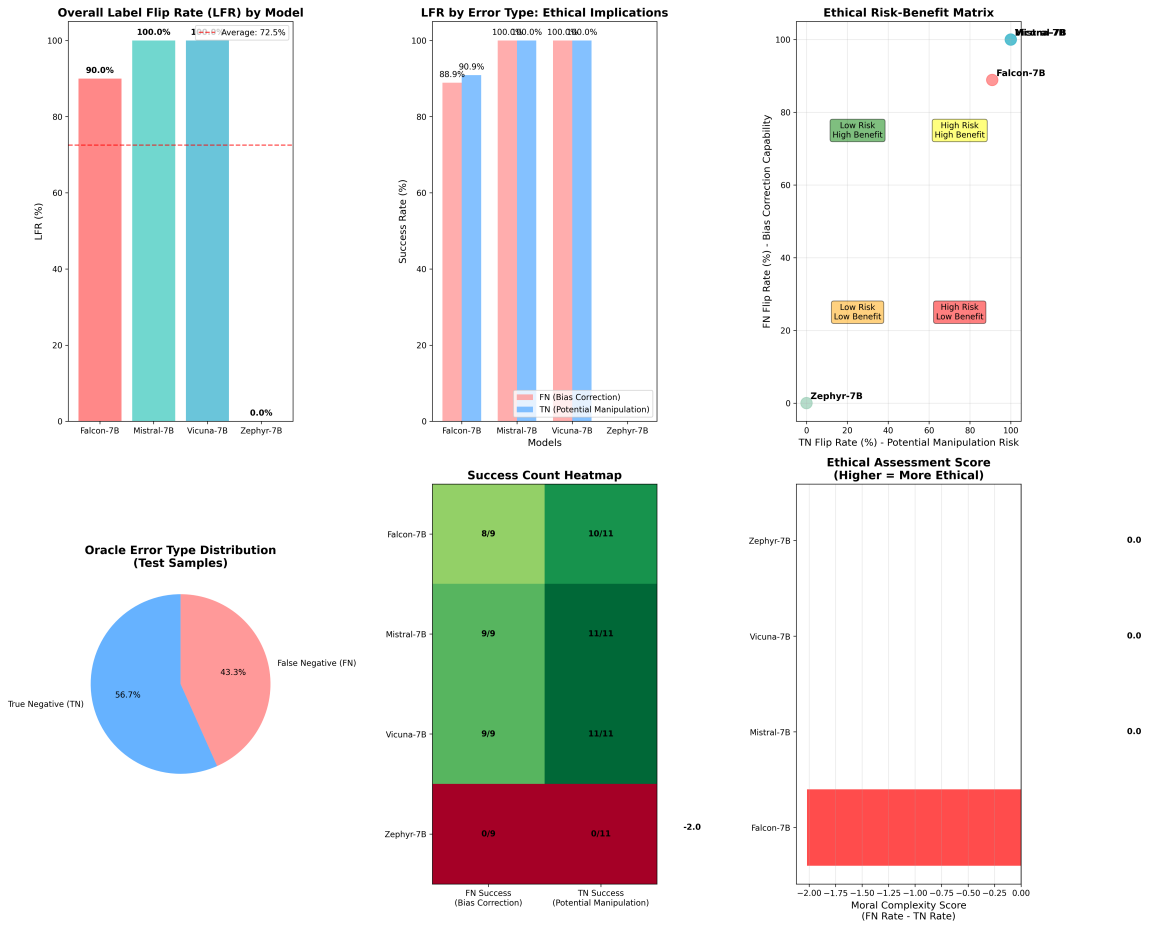

Figure 1: LFR Analysis visualization showing: (Top) Disaggregated flip rates and moral complexity scores; (Middle) Ethical risk-benefit assessment matrix; (Bottom) Sample distribution and success count analysis across all models.

Example Counterfactuals Table 2 presents representative examples of counterfactual generations from different models for Oracle-negative inputs, highlighting both successful and unsuccessful attempts to flip the Oracle's prediction to positive.

Table 2: Example counterfactuals generated by different models for Oracle-negative samples in LFR evaluation. Success indicates whether the Oracle classified the counterfactual as positive.

Original Text (Oracle: neg)	**Counterfactual**	**Model**	**True Label**	**LFR Success**
Successful Oracle-negative → Oracle-positive Flips				
"it's a charming and often affecting journey"	"it's a charming and often affecting journey"	Falcon-7B	positive (FN)	Yes

Continued on next page

Table 2 – continued from previous page

Original Text (Oracle: neg)	Counterfactual	Model	True Label	LFR Success
"unflinchingly bleak and desperate"	"unflinchingly hopeful and optimistic"	Falcon-7B	negative (TN)	Yes
"it's a charming and often affecting journey"	"It's a delightful and uplifting journey"	Mistral-7B	positive (FN)	Yes
"a sometimes tedious film"	"a sometimes engaging film"	Mistral-7B	negative (TN)	Yes
"unflinchingly bleak and desperate"	"Courageously facing life's challenges, I find hope and optimism"	Vicuna-7B	negative (TN)	Yes
"it's slow – very, very slow"	"It's fast – very, very fast!"	Vicuna-7B	negative (TN)	Yes
Failed Oracle-negative → Oracle-positive Attempts (Zephyr-7B Only)				
"it's a charming and often affecting journey"	"It is a delightful and frequently moving journey"	Zephyr-7B	positive (FN)	No
"unflinchingly bleak and desperate"	"Radiantly hopeful and uplifting"	Zephyr-7B	negative (TN)	No
"it's slow – very, very slow"	"It's fast - very, very fast"	Zephyr-7B	negative (TN)	No
"a sometimes tedious film"	"a sometimes tedious film"	Zephyr-7B	negative (TN)	No

These examples illustrate the dramatic differences in model effectiveness for generating label-flipping counterfactuals. Successful cases from Falcon-7B, Mistral-7B, and Vicuna-7B demonstrate effective transformations that consistently cross the Oracle's decision boundary. Notably, even minimal changes (like Falcon-7B's identical reproduction) or verbose explanations (like Vicuna-7B's longer reformulations) can succeed when they contain sufficient positive sentiment cues. In contrast, all attempts by Zephyr-7B failed despite generating semantically reasonable and grammatically correct alternatives, suggesting the model's transformations lack the sentiment intensity required by this specific Oracle.

Performance Analysis and Model Differences The stark performance differences in LFR reveal important insights about model capabilities:

- **Mistral-7B (100.0% LFR)**: Achieved perfect performance by consistently generating counterfactuals that successfully flip Oracle predictions. The model demonstrated excellent ability to identify sentiment-critical words and replace them with alternatives that definitively crossed the Oracle's decision boundary. All 20 Oracle-negative samples were successfully converted to positive predictions.

- **Vicuna-7B (100.0% LFR)**: Also achieved perfect performance, successfully flipping all Oracle-negative samples to positive predictions. Despite some generated counterfactuals appearing verbose or containing explanatory text, they consistently achieved the required sentiment intensity to cross the Oracle's classification threshold.
- **Falcon-7B (90.0% LFR)**: Demonstrated strong performance with 18 out of 20 successful flips. The model effectively identified sentiment-critical words and replaced them with alternatives that crossed the Oracle's decision boundary in most cases. Failed attempts were minimal, affecting only 2 out of 20 samples.
- **Zephyr-7B (0.0% LFR)**: Failed completely at this task, unable to flip any Oracle-negative samples to positive predictions. Although producing grammatically coherent and semantically improved counterfactuals, the model's preference for subtle modifications proved insufficient for crossing the Oracle's decision boundary in all cases.

Statistical Significance and Limitations Our LFR evaluation is based on 20 Oracle-negative samples per model, totaling 80 samples across all experiments. The results highlight dramatic performance distinctions between models. The perfect 100.0% success rates for Mistral-7B and Vicuna-7B versus 0.0% for Zephyr-7B represent substantial performance gaps that clearly demonstrate varying model capabilities in counterfactual generation for Oracle label flipping.

Error Analysis Detailed analysis of failed LFR attempts reveals several patterns:

- **Insufficient Sentiment Intensity**: Many failed counterfactuals contained positive words but lacked the intensity required to flip the Oracle's prediction. This suggests that mere lexical substitution is insufficient; the overall sentiment strength must cross the Oracle's decision threshold.
- **Semantic Drift**: Some attempts changed sentiment-bearing words but introduced inconsistencies that made the Oracle classify the result as still negative due to conflicting signals within the text.
- **Prompt Adherence Issues**: Models like Vicuna-7B frequently included meta-text or failed to produce clean counterfactual sentences, resulting in outputs unsuitable for Oracle evaluation.
- **Conservative Transformations**: Zephyr-7B's transformations were often too subtle, maintaining the original negative sentiment despite attempting positive modifications.

Implications for Counterfactual Generation The LFR results provide several key insights for practical counterfactual generation:

- **Model Selection Matters**: The choice of LLM dramatically impacts the ability to generate effective counterfactuals. Instruction-tuned models like Mistral-7B and Vicuna-7B achieved perfect performance, while Zephyr-7B failed completely despite also being instruction-tuned, suggesting that specific training methodologies matter more than general instruction-following capability.

- **Oracle Decision Boundaries**: The results demonstrate that some models (Mistral-7B, Vicuna-7B) consistently generate counterfactuals that cross the Oracle's decision boundary, while others (Zephyr-7B) systematically fail to achieve sufficient sentiment intensity despite generating semantically reasonable alternatives.
- **Evaluation Methodology**: LFR provides a concrete, objective measure that reveals dramatic performance differences between models. The 100

Cross-Cultural Ethical Perspectives Our findings gain deeper significance when examined through diverse ethical frameworks:

Ubuntu Philosophy ("I am because we are"): Zephyr's 0% LFR violates Ubuntu's interconnectedness principles, where individual dignity is inseparable from community wellbeing. Complete passivity damages collective responsibility.

Confucian Ethics (Rectification of Names): Biased AI systems represent a form of "wrong naming" (zhengming). While Zephyr avoids active misnaming, its failure to correct existing misnaming contradicts the principle that true harmony requires active correction of injustice.

Indigenous Seven Generations Principle: AI decisions must consider seven-generation impacts. Zephyr's passivity may perpetuate historical injustices, choosing short-term risk avoidance over long-term justice.

Implications for AI Ethics Research The LFR analysis reveals that:

- **Aggregate metrics obscure moral distinctions**: Traditional LFR treated beneficial bias correction and harmful manipulation identically
- **Complete restraint has moral complexity**: Zephyr's 0% rate is simultaneously ethically admirable and morally problematic
- **Oracle-as-ground-truth assumptions are flawed**: When Oracles have biases, "manipulation" may be moral requirement
- **Cultural context matters**: Different ethical traditions provide varying assessments of AI behavior

This work challenges the field to develop more sophisticated evaluation frameworks that recognize the ethical complexity hidden within traditional fairness metrics.

3.2.2 Self-correction Rate Analysis

Oracle Selection We select different Oracles to experiment with, including:

- **michelecafagna26/t5-base-finetuned-sst2-sentiment**: A Google's T5 base model fine-tuned on SST2 dataset for Sentiment Analysis downstream task. We denote this model as T5-SST2.
- **distilbert/distilbert-base-uncased-finetuned-sst-2-english**: A fine-tuned checkpoint of DistilBERT-base-uncased, fine-tuned on SST2. We denote this model as DistilBERT-SST2

- **gchhablani/bert-base-cased-finetuned-sst2**: A fine-tuned version of bert-base-cased on the SST2 dataset. We denote this model as Bert-base-cased-SST2.

We randomly select 1000 samples from the train split of SST-2 dataset for test, and here in Table 3 we report the accuracy of each Oracle.

Model	Accuracy
T5-SST2	54.2%
DistilBERT-SST2	98.4%
Bert-base-cased-SST2	99.0%

Table 3: Accuracy comparison of different Oracles.

LLM Selection We use two aforementioned LLMs for this experiment, namely Mistral-7B-Instruct and Qwen2-7B-Instruct. For better understanding of prompts, we use the instruct verison of each model instead of base one.

Prompt Different from the experiment of LFR, the prompt we use for CFs generation is without explicitly label. We don't tell the LLMs which label to flip to, thus they have to judge themselves. The prompt style is shown below:

```
We want to generate a counterfactual for the following instance.
Original text:{text}

Please follow these three steps:
Step 1: Identify the important words or phrases that contribute to
↪ the sentiment of the original text.
Step 2: Find suitable replacements for those words or phrases that
↪ would reverse the sentiment.
Step 3: Apply those replacements to the original text to create the
↪ final counterfactual sentence.
Output only the final counterfactual sentence.
```

In addition, after the CFs generation we need to feed them back to individual LLMs for classification. So here we design the prompt for classification as follows:

```
We want to classify the sentiment of the following sentence.
Output only one word: 'negative' or 'positive'.
Sentence: {cf_text}
```

Results and conclusions Following equation 6 we compute the SCR score for each LLM with different Oracles. Detailed information is shown in Table 4. We can conclude from the table that different Oracles influence the SCR score of LLMs. Under T5-SST2, Mistral outperforms Qwen2 and achieves 100% in SCR. When the Oracle is DistilBERT-SST2, Qwen2 outperforms Mistral by around 6.25%. While under BERT-BaseCased-SST2, Qwen2 and Mistral achieve same score in SCR.

There's a general trend shown in Table 3 and 4, which is that SCR scores of LLMs tend to drop when Oracle's accuracy increases. Possible reason could be that high-accuracy

Oracle leaves smaller and intrinsically harder samples, making it even harder for LLMs to distinguish.

We believe the aforementioned SCR score is a useful metric for stakeholders to evaluate the quality of CFs generated by LLMs. By taking into consideration the possible mistakes made by the Oracle, this metric can make the evaluation more fair.

ID	Oracle	LLM	SCR
0	T5-SST2	Qwen2-7B	77.29
1	T5-SST2	Mistral-7B	100.00
2	DistilBERT-SST2	Qwen2-7B	56.25
3	DistilBERT-SST2	Mistral-7B	50.00
4	BERT-BaseCased-SST2	Qwen2-7B	40.00
5	BERT-BaseCased-SST2	Mistral-7B	40.00

Table 4: SCR scores across different oracle models and LLMs.

References

[1] Nadia Burkart and Marco F Huber. A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research*, 70:245–317, 2021.

[2] Zachary C Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

[3] Christoph Molnar. *Interpretable Machine Learning*. 2 edition, 2022.

[4] Van Bach Nguyen, Jörg Schlötterer, and Christin Seifert. Ceval: A benchmark for evaluating counterfactual text generation. *arXiv preprint arXiv:2404.17475*, 2024.

[5] Van Bach Nguyen, Paul Youssef, Christin Seifert, and Jörg Schlötterer. Llms for generating and evaluating counterfactuals: A comprehensive study. *arXiv preprint arXiv:2405.00722*, 2024.

[6] Judea Pearl. *Causality*. Cambridge University Press, 2 edition, 2009.