Can we trust LLM Self-Explanations for Entity Resolution?

Tommaso Teofili
Roma Tre University, Elastic
tommaso.teofili@uniroma3.it

Donatella Firmani
Sapienza University of Rome
donatella.firmani@uniroma1.it

Nick Koudas
University of Toronto
koudas@cs.toronto.edu

Paolo Merialdo
Roma Tre University
paolo.merialdo@uniroma3.it

Divesh Srivastava
AT&T Chief Data Office
divesh@research.att.com

ABSTRACT

Large Language Models (LLMs) have recently shown strong performance on Entity Resolution (ER). Additionally, akin to their prowess in providing accurate predictions, these models often generate self-explanations alongside their predictions through prompting. While such self-explanations are appealing due to their negligible computational cost, their actual reliability remains largely unexplored. In this paper, we present the first large-scale systematic evaluation of LLM self-explanations for ER, focusing on feature attribution and counterfactual explanations at both the attribute and token levels. Across three LLMs, ten datasets, and multiple prompting strategies, we show that self-explanations are often unstable, weakly faithful, and poorly aligned with counterfactual evidence, revealing a substantial gap between plausibility and causal relevance. We further demonstrate that established post-hoc explanation methods provide significantly higher trustworthiness, but at a prohibitive computational cost when applied to LLMs. To bridge this gap, we introduce Ellmer, a hybrid explanation framework that leverages self-explanations as priors to guide post-hoc exploration. Ellmer achieves explanation quality comparable to post-hoc methods while reducing cost by up to an order of magnitude.

PVLDB Reference Format:
Tommaso Teofili, Donatella Firmani, Nick Koudas, Paolo Merialdo, and Divesh Srivastava. Can we trust LLM Self-Explanations for Entity Resolution? . PVLDB, 19(11): 3538 - 3551, 2026.
doi:10.14778/3836663.3836707

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/tteofili/ellmer.

1 INTRODUCTION

Entity Resolution (ER) is the fundamental problem of matching different representations of the same entity – e.g., a book or a customer record – across heterogeneous data sources, such as different review websites or corporate databases. As the usage of large, diverse datasets grows, ER has become an essential step in data preparation, enabling the integration and creation of new business-ready datasets. Recent advances in deep learning have been leveraged to tackle the ER problem, leading to significant improvements in accuracy and performance [1, 14, 17, 29, 31, 43, 44, 66–68]. However, these techniques rely on large volumes of training data, specifically labeled pairs of records, where each pair is annotated as either a match or a non-match.

Given the demonstrated power of Large Language Models (LLMs) and their considerable success across various domains – particularly in zero-shot and few-shot learning – recent studies have investigated their potential to tackle the ER problem, producing impressive results [29, 42, 45, 51, 61, 65].

Traditional ER systems based on deep learning work as closed models whose inner working mechanisms are opaque, and they can sometime make the right decisions for the wrong reasons [12, 56]. To overcome this limitation, several post-hoc methods have been proposed to generate explanations for the predictions of such ER systems [4–7, 12, 54, 55, 60].

LLMs are also opaque. However, along with their ability to produce precise predictions, they exhibit a unique trait: the ability of generating *self-explanations* [35] together with their ER outcomes.

Our work embarks on a thorough investigation to assess the quality and scalability of self-explanations and post-hoc explanations for ER predictions generated by LLMs. In particular, we analyze *feature attribution* and *counterfactual* explanations [26, 36], which are widely used in ER systems [20, 57] and exhibit high usability [48, 62]. The former assigns an *attribution score* to each feature in the input pair, indicating its contribution to the prediction; the latter provides examples of value changes that can alter the prediction.

In particular, we report the results of experimental evaluations which aim to examine three key research questions (RQ):

RQ-1: Are self-explanations trustworthy? Prior work has shown that LLM outputs may exhibit sensitivity to prompt variations across tasks [34, 70, 72], raising concerns about the stability of their generated explanations. We therefore assess the *robustness of the self-explanations* for the ER task under different prompting strategies. A further concern is the occurrence of "hallucinations", which in our context refer to the generation of self-explanations that appear plausible but are not grounded in the actual behavior of the LLM. Regrettably, such undesired behaviors could undermine the *reliability of self-explanations*, thus hampering the effectiveness of ER predictions with self-explanations. To assess this issue, we conduct an experimental evaluation using established metrics from the explainable AI literature: faithfulness [3] for feature attribution, and validity [40] for counterfactual self-explanations. While faithfulness and validity do not capture all dimensions of explanation quality, they are widely adopted proxies for assessing whether explanations are grounded in the actual decision logic of a model [19].

Proceedings of the VLDB Endowment, Vol. 19, No. 11 ISSN 2150-8097.
doi:10.14778/3836663.3836707

Also, we assess the consistency between attribute level and token level explanations, and the agreement between feature attribution and counterfactual explanations.

RQ-2: Are post-hoc explanations more trustworthy? Given the availability of post-hoc explanations for black-box ER systems, we wonder whether they are more reliable than self-explanations. To this end, our investigation proceeds comparing the self-explanations of LLMs to the post-hoc explanations provided by systems specifically developed for ER. Our comparison encompasses both feature attribution and counterfactual explanations, and involves the SOTA systems CERTA [54, 55], LEMON [6], and MINUN [60]. All these methods are model-agnostic and thus can be applied to LLMs as well.

RQ-3: Is it feasible to compute post-hoc explanations for LLMs? Our experiments unveil an intriguing trade-off: post-hoc explanations exhibit significantly higher reliability compared to self-explanations. However, this reliability comes at a computational cost, as post-hoc methods require additional processing steps. To address the trade-off between reliability and efficiency, we propose ELLMER, a hybrid explanation method that seamlessly integrates the inherent efficiency of self-explanations with the trustworthiness of post-hoc explanations to enhance reliability. This approach not only mitigates the risk of hallucinations but also ensures scalability, making it a practical solution for real-world ER applications.

To address the above research questions, we conduct an experimental evaluation using three LLMs and ten datasets, allowing us to assess the robustness and generality of the proposed analysis across heterogeneous settings.

In summary, this paper makes the following contributions:

- We systematically evaluate the trustworthiness of LLM-generated self-explanations for entity resolution, analyzing their sensitivity to prompting strategies and their susceptibility to hallucinations.
- We compare self-explanations and post-hoc explanations in terms of both effectiveness and computational efficiency in the ER setting.
- We experimentally assess a hybrid explainability approach for ER, ELLMER, which combines self- with post-hoc explanation methods to mitigate scalability limitations.
- We provide three new synthetic datasets for the ER task, that current LLMs have not previously had access to.

Outline Section 2 presents preliminary notions. Section 3 illustrates self-explanations, post-hoc and hybrid explanation methods. Section 4 reports experimental results. Section 5 discusses related work, and Section 6 concludes the paper.

2 FOUNDATIONS

We now introduce the core concepts underlying our analysis.

Entity Resolution Problem We consider the clean-clean entity resolution setting, where two data sources U and V are individually (almost) duplicate-free. Formally, given two sets of records U and V, the goal is to classify record pairs in $E = U \times V$. Given a pair of records $(u, v) \in U \times V$, we define $M(u, v) = y$ as the prediction generated by a model M; $y \in \{T, F\}$, where T indicates a *match*, that is, u and v refer to the same entity, whereas F indicates a *non-match*, that is, u and v refer to different entities.

Solving ER with LLMs We focus on models that are LLMs and thus we interact with them via prompting. In the prompt, we represent records as sequences of tokens $u = t_1, t_2, \ldots, t_k$ where the schema can be loosely specified, such as u ="Model: Canon EOS 5D Mark IV, Resolution: 30.4 megapixels, Sensor: Full-frame CMOS". For sake of simplicity, we use the symbol u both to refer to the record and the corresponding set of tokens $\{t_1, t_2, \ldots\ t_k\}$. In the prompt, we also add instructions on the ER task and on how to solve it, requiring the answer "yes" for a matching prediction and "no" for a non-matching prediction. Further details depend on the specific prompting strategy, which are detailed in Section 3.1.

Explanations for ER We consider *feature attribution* and *counterfactual* explanations, which are widely adopted for the ER task and can be evaluated by means of established metrics in the explainable AI literature [3, 40]. More specifically, in our context:

- a *feature attribution* explanation assigns an *attribution score* to each feature in $u \cup v$, capturing its contribution to the outcome of $M(u, v)$;
- a *counterfactual* explanation provides instructions on how to change features in $u \cup v$ in order to flip the outcome of $M(u, v)$.

Technically, a feature attribution explanation consists of a map that associates each feature with a numerical score, while a counterfactual explanation consists of a pair (u', v') that is equal to (u, v) except for one or few feature values.

Throughout this work we use the term *feature* as a unifying abstraction to refer to either of two granularities at which explanations can be produced: *attribute* or *token*. Following standard database terminology, an *attribute* denotes the complete value of a single field in a record (e.g., the entire `title` or `price` value), whereas a *token* denotes an individual word within an attribute value. Attribute-level explanations assign a single feature attribution score to each field and propose counterfactual modifications at the field level; token-level explanations operate at finer granularity, scoring and modifying individual words. The two granularities are not mutually exclusive, as they address different user needs [16, 32].

To give an example, consider the records u ="Model: Nikon D850, Sensor: Full-frame CMOS", and v ="Model: Nikon D750, Sensor: CMOS", with $M(u, v) = F$. A token-level feature attribution explanation might assume the form {D850→.8, D750→.6, Full-frame→.1, ...}, highlighting the importance of tokens such as "D850" and "D750". Similarly, an attribute-level feature attribution explanation might be {Model: Nikon D750→.7, Model: Nikon D850→.6, Sensor: Full-frame CMOS→.2 Sensor: CMOS→.01}, revealing that the value of the Model attribute in u is the most relevant for the prediction. A counterfactual token-level explanation could indicate that changing the token "D850" in u into "D750" would flip the prediction from F to T.

Faithful and actionable explanations. An explanation is *faithful* if it accurately reflects the model's decision process, as opposed to merely being plausible or coherent to a data steward or analyst reviewing the output [19, 32]. This distinction is relevant because a fluent, human-readable explanation can be compelling without necessarily corresponding to the features the model actually relied upon. For feature attribution explanations, faithfulness is operationalised through feature removal tests: a faithful ranking tends to assign high scores to features whose masking causes the largest degradation in model performance [11, 49]. Importantly, faithfulness is defined relative to the *model's* decision process rather than

to ground truth, so a faithful explanation for a wrong prediction is not necessarily a contradiction: it may accurately characterise *why the model erred*, which can be useful for auditing, error analysis, and targeted correction. This connection to *actionability* [24, 59] gives faithfulness practical relevance beyond its technical definition. We provide experimental evidence of this in Section 4.4.

3 SELF-EXPLANATION AND POST-HOC METHODS

This section introduces the explainable AI techniques for ER considered in our analysis. We distinguish between two main families of approaches: *self-explanations* and *post-hoc explanations*. Both aim at providing insight into ER predictions through feature attribution and counterfactual explanations. Then, we introduce a *hybrid* approach, which combines self- with post-hoc explanations.

Self-explanation approaches rely on large language models that are prompted to jointly produce predictions and explanations within the same model invocation, explicitly clarifying the rationale underlying their decisions. This paradigm, introduced in [50], is discussed in Section 3.1, where we describe how different prompting strategies are tailored to the ER task.

Post-hoc approaches, instead, generate explanations after predictions have been produced, by analyzing the behavior of the model through targeted perturbations of input features and observing the resulting changes in its outputs [37, 52]. Section 3.2 illustrates the SOTA post-hoc explanation methods adopted in our analysis.

The hybrid approach aims to leverage the strengths of both self- and post-hoc explanations, and it is presented in Section 3.3.

3.1 Self-Explanations

Thanks to the flexibility granted by prompting [9, 33], LLMs can be instructed to return different forms of self-explanations. We are interested in obtaining structured responses representing feature attribution and counterfactual explanations. Based on previous work [9, 42, 63], we leverage three different prompting approaches: *Zero-Shot*, *In-Context Learning*, *Chain-of-Thought*.

Feature attribution explanations. Across all three prompting strategies, the LLM is asked to assign a numerical score to each input feature, at either attribute or token granularity, reflecting how much that feature is perceived to contribute to the matching decision. The prompt specifies the output format (a score per feature, returned as a structured JSON object) and provides a natural-language definition of what the score represents, but imposes no constraint on how scores should be derived.

Counterfactual explanations. For counterfactuals, the model is asked to produce a modified version of the input pair such that the predicted label would flip: a non-match pair becomes a match, or vice versa. As with feature attribution explanations, the prompt defines the target output structure (a record pair with one or more feature values altered) and leaves the construction strategy to the model. The expected behavior (minimal and semantically coherent edits that are sufficient to reverse the prediction) is implied by the task description.

After initialization with a short description of the ER task, the three approaches work as follows.

Zero-Shot (ZS) As described in [30], we rely on the simple and cost-effective zero-shot prompt, where no prior context is given to the LLM about other records from the dataset. The only context we provide is a textual description of what feature attribution and counterfactual explanations consist of. Figure 1 shows an example of a ZS prompt. First, the prompt illustrates to the LLM what feature attribution and counterfactual explanations are. Then, it prescribes to the LLM the expected output as a JSON object with *(i)* a yes/no response indicating the matching/non-matching prediction, *(ii)* a feature attribution explanation table having a column for each input feature containing each feature's score, *(iii)* a counterfactual explanation table having a column for each input feature containing its counterfactual value. Finally it provides the pair of records ($record1$ and $record2$) to be processed.

```
SYSTEM
You are an expert assistant for Entity Resolution tasks.
USER
A feature attribution highlights the most relevant features that contributed
to the prediction. Positive scores support the predicted class; negative
scores are evidence against it; absolute value indicates strength.

A counterfactual explanation describes an altered input — with one or more
key features changed — such that the prediction would flip.

Given the pair of records below, predict whether they refer to the same
real-world entity. Return ONLY strict JSON (no markdown, no extra text):
  { "prediction": "yes"|"no",
    "feature_attribution": { "ltable_<attr>": <score>, "rtable_<attr>": ... },
    "counterfactual_explanation": { "ltable_<attr>": "<new value>", ... } }
USER
record1:  title=The Whispering Shadows  |  author=Alice Johnson  |  genre=Mystery
          publication_year=2021  |  plot=A detective unravels a series of murders
          linked to an ancient curse in a small coastal town.

record2:  title=Whispering Shadows  |  author=A. Johnson  |  genre=Mystery
          publication_year=2021  |  plot=A detective story involving murders
          and a mysterious old curse near the coast.
```

Figure 1: Example of a Zero-Shot prompt.

```
SYSTEM
You are an expert assistant for Entity Resolution tasks.
USER
A feature attribution highlights the most relevant features that contributed
to the prediction. Positive scores support the predicted class; negative
scores are evidence against it; absolute value indicates strength.

A counterfactual explanation describes an altered input — with one or more
key features changed — such that the prediction would flip.

Think step by step: compare each field one by one, note agreements and conflicts.
After your reasoning, return ONLY strict JSON (no markdown, no extra text):
  { "prediction": "yes"|"no",
    "feature_attribution": { "ltable_<attr>": <score>, "rtable_<attr>": ... },
    "counterfactual_explanation": { "ltable_<attr>": "<new value>", ... } }

record1:  title=The Whispering Shadows  |  author=Alice Johnson  |  genre=Mystery
          publication_year=2021  |  plot=A detective unravels a series of murders
          linked to an ancient curse in a small coastal town.

record2:  title=Whispering Shadows  |  author=A. Johnson  |  genre=Mystery
          publication_year=2021  |  plot=A detective story involving murders
          and a mysterious old curse near the coast.
```

Figure 2: Example of a Chain-of-Thought prompt.

Chain-of-Thought (CoT) Using the Chain-of-Thought prompting technique [63], we prompt the LLM to generate explicit intermediate reasoning steps leading to its final prediction for the entity resolution explanation task. In contrast to the zero-shot setting, where predictions and explanations are produced directly without an explicit reasoning structure, CoT encourages the model to decompose the decision process into successive steps, making the resulting explanations more structured and transparent. Figure 2 provides an example of a CoT prompt.

In-Context Learning (ICL) Consistent with prior work showing that In-Context Learning can improve LLM performance on downstream tasks [9], we augment the ZS prompt with a set of labeled examples. Specifically, we provide ten examples (five matching and five non-matching pairs) each accompanied by feature attribution and counterfactual explanations, allowing the model to condition its predictions and explanations on observed patterns. Figure 3 reports a portion of the examples that we provide in an ICL prompt.

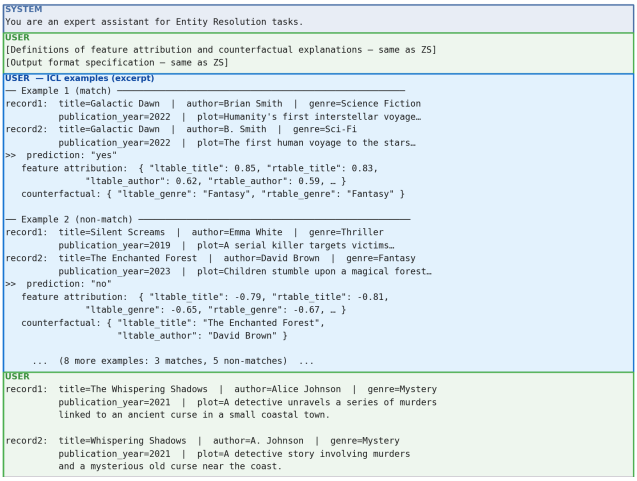


Figure 3: Example of an In-Context Learning prompt.

3.1.1 Eliciting vs. Instructing Explanations. A key design choice in our evaluation of self-explainers is that we constrain the prompts to specify *what kind* of explanation the model should produce, a ranked feature attribution over attributes or tokens, and a minimal counterfactual record pair, but deliberately leave unspecified *how* that explanation should be computed. This choice reflects the nature of the object being evaluated: a self-explanation is not the output of an explicit algorithmic procedure invoked by the user, but rather what the LLM produces *naturally* as a complement to its matching prediction. Imposing a specific computational method in the prompt would shift the task from *eliciting* an explanation to *instructing* one, conflating the model's native reasoning with externally prescribed procedure. Instructing the model on *how* to compute its explanations would change the object of study: we would be measuring the model's ability to execute a prescribed procedure, not the reliability of the explanations it produces autonomously as part of its inherent prediction workflow.

3.2 Post-hoc explanations

Post-hoc methods for the ER task include Mojito [12], Explainer [13], Landmark [5], Minun [60], certa [54], and Lemon [6].

Mojito, Landmark, and Lemon build upon the seminal LIME approach [47]. In our experimental analysis, we use certa, Lemon, and Minun as post-hoc explanation methods as they represent among the most recent and reliable solutions specifically designed for the ER task, and have been evaluated on standard benchmarks.[1] certa and Lemon generate feature attribution explanations, certa and Minun provide counterfactual explanations, enabling a direct comparison with self-explanations when addressing the research questions on trustworthiness, effectiveness, and efficiency. Lemon [6] generates perturbed variants of the input pair (u, v) by masking or replacing individual features, queries M on each variant, and fits a weighted linear surrogate model g on the resulting (perturbation, prediction) pairs using a LIME-style proximity kernel. The attribution score of each feature is the corresponding coefficient in g, reflecting how much that feature's presence influences the local linear approximation of M's decision boundary around (u, v).

certa [54] produces both feature attribution and counterfactual explanations via a probabilistic perturbation framework. certa builds on the concept of *open triangles* over (u, v) by finding another record $w \in U$ (called *support record*) such that the ER system predicts (w, v) to have a different prediction with respect to $M(u, v)$. *Perturbing* u by progressively copying attribute values from w to u, deriving a u', increasingly making u' more similar to w based on their content, at some point the prediction of the model will flip, declaring u' and v such that $M(u, v) \neq M(u', v)$. Repeating the same procedure for many support records produces evidence of the influence that attributes and sets of attributes have on the input prediction. For feature attribution explanations, the attribution score of an attribute is defined as the probability that changing the value of that attribute is a *necessary* factor for flipping the outcome of the prediction (*probability of necessity*), across different open triangles. Symmetrically, certa generates *counterfactual* explanations having the highest probability that changing the value of a certain *set of attributes* is a *sufficient* factor for flipping the outcome of a prediction (*probability of sufficiency*). Such probabilities are computed by a frequentist approach. The number of times an attribute is changed over the number of actual flips gives the *probability of necessity*, the number of times changing a set of attributes results in a flip over the number of times that the set of attributes is changed gives the *probability of sufficiency*.

Minun [60] is a post-hoc method for the ER task aimed at generating counterfactual explanations. It adopts an approach based on perturbations, which are produced by modifying the input pair until the original prediction flips. The perturbation is performed by considering all possible sequences of token insertions, removals, and replacements to transform one record (e.g., u in (u, v)) into the other. Greedy and approximate algorithms efficiently compute the shortest transformations.

3.3 Ellmer: a Hybrid Explanation Method

Post-hoc explanations require performing a large number of predictions over the perturbed samples. In the LLM scenario, this yields significant computational (and monetary) cost because each prediction corresponds to a call to a LLM. On the other hand, self-explanation methods can generate explanations at the cost of a single prediction but their truthfulness is not guaranteed. Although self-explanations can highlight seemingly reasonable features, there is no guarantee that these are the features that truly influenced the prediction by the LLM.

For these reasons, we introduce a hybrid method, Ellmer, which leverages the strengths of both self- and post-hoc explanations.

The main intuition of Ellmer is to use self-explanations as a prior to optimize search for flip-producing feature changes in the perturbation space. Reducing the number of explored perturbations can directly reduce the number of predictions (and thus the computational cost) without compromising on trustworthiness.

Given a prediction $M(u, v) = y$ produced by an LLM M, let $P = \{p_1, \ldots, p_n\}$, be a set of prompts used for generating self-explanations. For each $p \in P$, we collect the top k features of feature attribution self-explanations into a list Ξ. In our implementation, P includes the three prompts (ZS, ICL, CoT) described in Section 3.1. We empirically observed that adding other prompts

[1] We extend the original implementation of certa[54], which is limited to attribute-level explanations, to support explanations at both the attribute and token levels.

(e.g., using different forms of CoT and examples in ZS) does not produce significant improvements.

In the worst case, Ξ may contain up to $n \cdot k$ features, when each of the n self-explanations yields a disjoint set of top-k features. To bound the size of Ξ, we apply a frequency-based selection strategy: we count how often each feature appears across the n self-explanations and retain only the k most frequent ones, thereby favoring agreement across different prompts.

We call Ξ^* the final set of features (attributes or tokens) selected from Ξ. This set is used as a *perturbation mask* to be passed to post-hoc algorithms. Note that any post-hoc algorithm can be used to generate a feature attribution explanation, a counterfactual explanation or both, provided that it can be made to focus only on the features specified in Ξ^*, to reduce the search space.

To avoid cases in which the total attribution scores of the selected feature set Ξ^* is negligible (indicating the absence of sufficiently informative features), we require the sum of attribution scores over Ξ^* to exceed a threshold ϕ. If this condition is not satisfied, we iteratively increase k until either the threshold ϕ is met or Ξ^* expands to include all available features. The values of ϕ and k are determined through empirical evaluation. In our experiments, we set $\phi = 0.5$, $k = \frac{|\text{features}|}{2}$ for attribute-level explanations, and $k = 5$ for token-level explanations, as these values provided a good trade-off between coverage and compactness.

In our experimental evaluation, we instantiate Ellmer with different post-hoc explanation methods depending on the explanation type. For feature attribution explanations, we employ Lemon and certa, while for counterfactual explanations we use Minun and certa. Throughout the paper, we denote these instantiations as Ellmer$_L$, Ellmer$_C$, and Ellmer$_M$, corresponding to hybrid explanations built upon Lemon, certa, and Minun, respectively. For the attribute-level explanations, we populate Ξ^* with attribute values (e.g., "Nikon D750"), whereas for the token-level version we fill Ξ^* with individual tokens (e.g., "Nikon").

4 EXPERIMENTS

In this section we discuss experimental results related to the research questions introduced in Section 1.

LLMs and Prompting We consider three representative LLMs: Claude-4.6-Sonnet [2] (April 2026), ChatGPT-5.0 (December 2025), and LLaMA 3.1-8B (December 2025). These models were selected to capture complementary design choices in terms of architecture, scale, training regimes, and deployment settings, including both closed-source and open-weight models. This selection allows us to assess whether the observed behavior of self-explanations is consistent across heterogeneous LLM configurations rather than being specific to a single model. To ensure reproducibility of results, we set the temperature parameter to 0 for all models in all reported experiments. This reduces stochastic variations in the LLMs, making the results comparable across runs. For each LLM, self-explanations are generated using the prompting strategies described in Section 3.1, enabling a systematic comparison of the robustness of explanations under different prompting conditions. Whenever possible, we leverage *optimized batch prompting* [22], to mitigate the inherent latency amplification in LLMs inference.

Datasets We evaluate entity resolution predictions produced by the considered LLMs on selected datasets from the DeepMatcher [41] and WDC [46] benchmarks, which are widely used for semantic entity resolution. In particular, we consider the Abt-Buy (AB), BeerAdvo-RateBeer (BR), Fodors-Zagats (FZ), Walmart-Amazon (WA), and Amazon-Google (AG) datasets from DeepMatcher,[2] and the Cameras and Watches datasets from WDC.[3]

Since these datasets may have been observed during LLM pretraining, we additionally introduce two novel synthetic datasets, consisting of descriptions of *fictional books* (FB) and *fictional car parts* (FCP), generated by an independent LLM (Mistral Nemo [23]). Each dataset consists of two table sources L and R, a labeled training set G, and a labeled test set T, with an overall target size s. Initially, both sources are empty, i.e., $L = \emptyset$ and $R = \emptyset$. Then, the datasets are generated according to the following procedure:

(1) For a given domain, we use Mistral to generate two disjoint sets of entity descriptions, A and B, with $A \cap B = \emptyset$.
(2) From A, we randomly select one third of the entities and generate alternative textual representations that preserve the same underlying identity, obtaining a set C.
(3) Positive (matching) pairs are generated by sampling $\langle a, c \rangle$, with $a \in A$ and $c \in C$, and are evenly split between the training set G and the test set T (label = 1).
(4) Negative (non-matching) pairs are generated by sampling $\langle a, b \rangle$, with $a \in A$ and $b \in B$, and are evenly split between G and T (label = 0).
(5) The two sources are updated as $L = L \cup A$ and $R = R \cup B \cup C$.
(6) The procedure is repeated until $|L| + |R|$ reaches s.

For the FB dataset, we additionally generate several batches of samples in languages other than English (Spanish, German, and Italian) to increase linguistic diversity and overall dataset difficulty.

We also construct a synthetic ER dataset (dubbed *FakER*), designed as a worst-case scenario to assess scalability and robustness (rather than realism), to evaluate explanations generated under high attribute dimensionality. Each source contains 300 records, comprising 150 shared entities appearing in both sources and 150 source-specific entities unique to each source. Shared entities form the positive matches across sources, while source-specific entities act as natural distractors. All records follow a common schema with 105 textual attributes and a unique source-specific identifier. Attribute values are entirely textual and generated using a random mixture of realistic data types, including personal names, postal addresses, company names, short natural-language sentences, and short free-text fragments. This design reflects ER scenarios in which entities are described by large numbers of heterogeneous textual fields, rather than a small set of curated attributes. To simulate real-world data inconsistencies, we introduce controlled textual variations independently in each source. Starting from a clean base representation for each shared entity, we apply attribute-level noise including character-level typos, abbreviations and truncations, synonym substitutions (e.g., street -> st), missing or extra characters, and limited word reordering. The noise level is asymmetric across sources: medium variation in Source A and high variation in Source

[2] https://github.com/anhaidgroup/deepmatcher/blob/master/Datasets.md
[3] https://huggingface.co/datasets/wdc/products-2017

Table 1: Datasets details. For each dataset, we report: sizes of left and right tables ($|L|$ and $|R|$), total number of candidate pairs ($|L \times R|$), test set size($|test|$), number of attributes (*attr.*), average number of tokens per record (*tokens*), F1 score of ER predictions obtained with the three LLMs.

								F1-score		
	Dataset	|L|	|R|	$\|L \times R\|$	|test|	attr.	tokens	LlaMa3.1-8B	ChatGPT-5	Claude-4.6
DeepMatcher	AB	1,081	1,092	5,743	1,916	3	59	0.93	0.93	0.91
	BR	4,345	3,000	68	91	4	26	0.90	0.95	0.94
	FZ	533	331	110	189	6	37	1	1	1
	WA	2,554	22,074	962	2,049	5	15	0.90	0.93	0.80
	AG	1,363	3,226	1,167	2,293	3	10	0.88	0.95	0.82
WDC	Cameras	17,128	17,128	16,028	1,100	7	229	0.62	0.97	0.95
	Watches	22,721	22,721	21,621	1,100	7	193	0.59	0.96	0.95
Synthetic	FB	520	495	100	317	5	35	0.77	0.81	0.85
	FCP	500	600	220	163	8	37	0.74	0.79	0.80
	FakER	300	300	150	350	210	776	0.83	0.99	0.93

B, reflecting differences in data quality commonly observed in practice. All random processes are seeded to ensure reproducibility. Ground truth consists of 150 positive record pairs corresponding to shared entities and 200 randomly sampled negative pairs drawn from non-matching record combinations.

For each dataset, the LLMs are prompted on all record pairs in the test set to predict match/non-match labels. Table 1 reports the main features of the datasets and the F-1 measure of the predictions returned by the LLMs that we have used in the experiments.

Metrics The evaluation of explainability results aims to verify that the explanations accurately capture the features that truly influenced the prediction [21]. We use two popular trustworthiness metrics from the explainable AI literature: *faithfulness* [3] for feature attribution, *validity* [40] for counterfactual explanations.

The main idea behind the *faithfulness* metric is that modifying features with high attribution score should cause a significant change in the model predictions, whereas modifying features with low scores should have a negligible effect. For each dataset, we consider all record pairs and progressively remove, for each pair, a fraction θ of the *most important* features (attributes or tokens), according to their attribution scores. For instance, $\theta = 0.3$ means that the top 30% of features in the non-increasing attribution score ranking are removed. We then compute a curve that maps θ to the resulting prediction performance, measured in terms of F1-score. Faithfulness is defined as the area under this curve [3]. Highly trustworthy explanations are expected to yield a smaller area, as removing even a small fraction of highly important features should lead to a sharp degradation in prediction performance. In our experiments, we set $\theta \in \{0.1, 0.2, 0.33, 0.5, 0.7, 0.9\}$.

For counterfactual explanations, we assess their quality using the *validity* metric, which measures the proportion of proper counterfactuals. Specifically, for each dataset, we apply the changes prescribed by the counterfactual explanations to the corresponding record pairs and re-query the LLM. Validity is then computed as the fraction of cases in which the model's prediction is successfully flipped, either from match to non-match or vice versa.

All experiments are repeated five times and results are reported as means. Run-to-run variability is low (coefficient of variation <5% across all metrics and methods). Statistical significance is assessed using paired two-sided t-tests across datasets, comparing the mean performance over the five runs for each method. Holm–Bonferroni correction [18] is applied to control the family-wise error rate, with significance assessed at p<0.05.

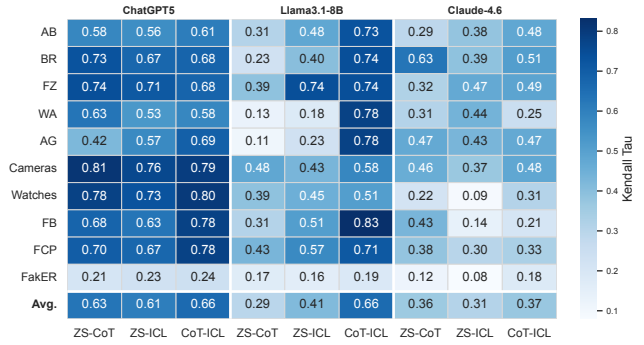
4.1 RQ-1: Are self-explanations trustworthy?

To address RQ-1, we analyze the trustworthiness of LLM-generated self-explanations for ER along two complementary dimensions. First, we investigate the sensitivity of self-explanations to different prompting strategies, assessing whether variations in prompts lead to unstable or inconsistent explanatory outputs. Second, we examine the extent to which self-explanations are affected by hallucinations and evaluate their impact on the reliability of explanations in the ER setting. Finally, we analyze the consistency between attribute and token-level explanations, and the agreement between feature attribution and counterfactual explanations.

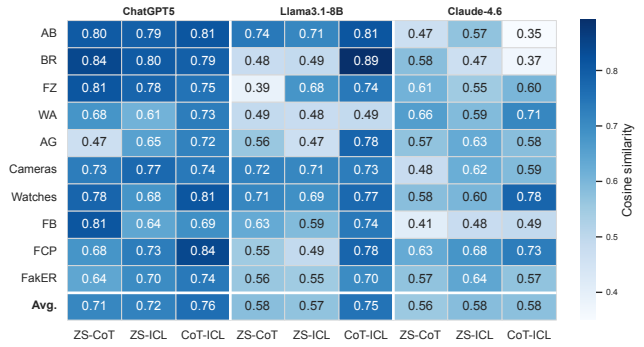
4.1.1 Sensitivity to Prompt Strategies. To assess prompt sensitivity in self-explanations, we first generate predictions and corresponding self-explanations for all record pairs considered in each dataset using the ZS, CoT, and ICL prompts. Then, for each prediction we measure the pair-wise similarity of the result of the three self-explanations strategies. Robust self-explanations should remain consistent even with different prompts.

We evaluate the similarity of explanations as follows. For feature attribution explanations, we consider the relative importance attributed by the LLM to each feature, i.e., attribute or token, because absolute scores are more likely to change. Specifically, for each self-explanation we consider the ranking of features by non-increasing attribution score, and measure the Kendall-Tau correlation coefficient [25] of rankings corresponding to different prompts. A Kendall-Tau correlation of 1 denotes complete agreement, whereas −1 indicates complete disagreement. A value of 0 corresponds to no correlation. For counterfactual self-explanations we consider instead textual similarity. First, we transform each self-explanation into a bag-of-words vector weighted with TF-IDF score. Then we measure the cosine similarity of vectors corresponding to different prompts for the same prediction.

For each dataset, Figure 4 reports the average similarity scores of *ZS* self-explanations vs. *CoT*, then *ZS* vs. *ICL*, and finally *CoT* vs. *ICL*. We report only attribute-level results as we observed that token-level results are very similar. In general, feature attribution explanations show suboptimal agreement, with Kendall-Tau often between 0 and 0.5, for Llama-3.1-8B and Claude-4.6-Sonnet. ChatGPT-5 shows a higher degree of agreement, across pairs, but still averaging a mild 0.6 KT. For ChatGPT-5 and Llama-3.1-8B, *CoT* and *ICL* seem to agree more steadily. Claude-4.6-Sonnet shows a lower degree of agreement for feature attribution explanations. Counterfactual explanations report high values of cosine similarity

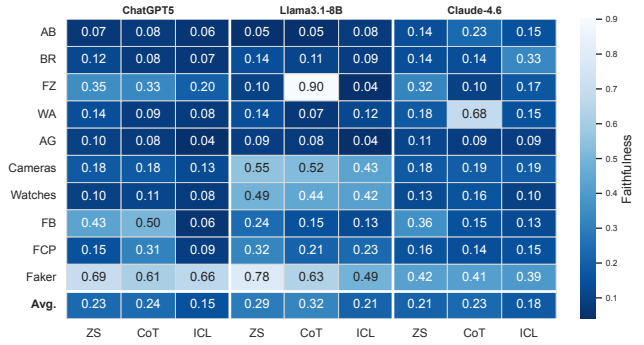


(a) Avg. Kendall-Tau correlation for feature attribution self-explanations (↑).

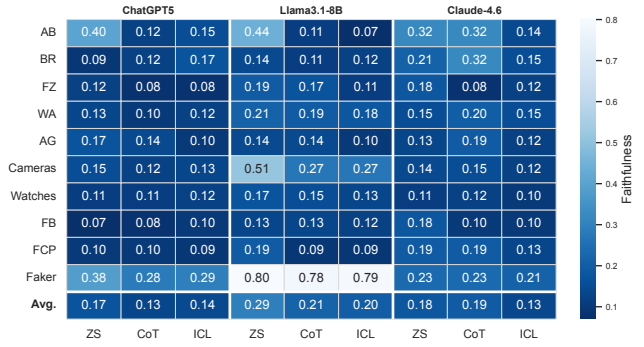


(b) Avg. cosine similarity for counterfactual self-explanations (↑).

Figure 4: Agreement of self-explanations (attribute-level); 1 (darkest) corresponds to the highest agreement (↑).



(a) Attribute-level (↓).



(b) Token-level (↓).

Figure 5: Faithfulness of feature attribution self-explanations. Values close to 0 (darker) are better (↓).

in some cases but also present high variability, suggesting that also counterfactual self-explanations are often in disagreement for the same prediction, when generated with different prompts. Also in this case we observe a higher degree of similarity between counterfactual explanations generated with *CoT* and *ICL* prompts for ChatGPT-5 and Llama-3.1-8B, whereas such similarity is generally lower for Claude-4.6-Sonnet.

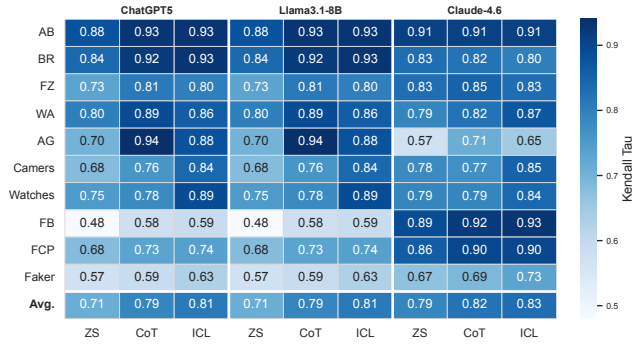
4.1.2 Faithfulness and Validity of Self-Explanations. We now provide results for a key desiderata for self-explanations, that is, their ability to capture the actual matching criteria used by the LLM.

Figure 5 reports the faithfulness results for all the datasets considered in our experiments at attribute and token level. With respect to prompting strategies, ICL consistently achieves better faithfulness than the other approaches across models, for both attribute-level and token-level explanations. An exception is observed for ChatGPT-5 at the token level, where CoT slightly outperforms ICL. Similar observations hold for counterfactual self-explanations, whose results are presented in Figure 6, but in this case the ICL strategy outperforms the others with all the models.

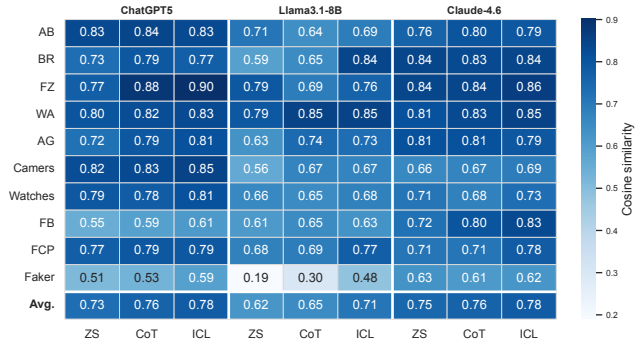
ChatGPT-5 consistently produces the most trustworthy feature attribution explanations, at both the attribute and token levels. In contrast, Claude-4.6-Sonnet produces counterfactual explanations with a higher validity; while generally worse, LLaMA3.1-8B still exhibits comparable performance for both feature attribution and counterfactual explanations.

Overall, counterfactual explanations tend to be slightly more trustworthy at the attribute level than at the token level, although the observed differences strongly depend on the specific dataset and the LLM considered. Datasets with a larger number of tokens (Cameras and Watches) do not display markedly different behaviors, suggesting that explanation quality does not scale proportionally with record length. All methods, however, struggle on the FakER dataset, where both faithfulness and validity substantially degrade.

The dominant pattern across models is that ICL and CoT tend to outperform ZS, yielding lower faithfulness and higher validity, but the statistical footprint of this ordering is both model and granularity-dependent. The effect of prompting strategy is concentrated at the attribute level. For both ChatGPT-5 and LLaMA-3.1-8B, CoT and ICL significantly improve faithfulness and validity relative to zero-shot prompting, whereas differences largely disappear at the token level. The strongest gains are observed for ChatGPT-5, where ICL substantially outperforms zero-shot prompting on both metrics (all ($p < 0.001$)). However, the additional benefit of ICL over CoT is limited: the ICL–CoT comparison is never significant at the token level and becomes significant only for ChatGPT-5 at the attribute level. Claude-4.6 exhibits a markedly different behavior. Across all twelve pairwise comparisons, only two reach statistical significance,

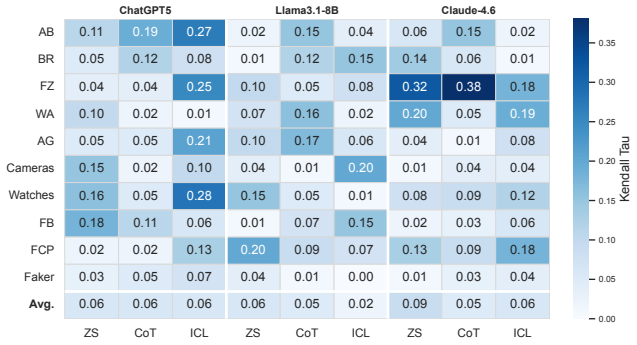


(a) Attribute-level (↑).

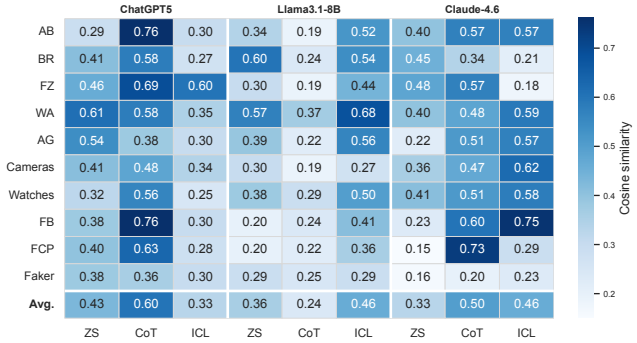


(b) Token-level (↑).

Figure 6: Validity of counterfactual self-explanations. Values close to 1 (darker) are better (↑).



(a) Kendall-Tau correlation between attribute and token-level self-explanations (↑).



(b) Cosine similarity between attribute and token-level counterfactual self-explanations (↑).

Figure 7: Cross granularity consistency of self-explanations.

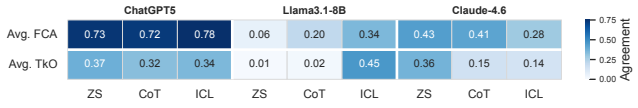
both favouring ICL over zero-shot prompting: token-level faithfulness and attribute-level validity. All remaining comparisons are non-significant, indicating that self-explanation quality is largely insensitive to the choice of prompting strategy. Overall, these results show that improved prompting can enhance self-explanation quality, particularly for ChatGPT-5 and LLaMA-3.1-8B at the attribute level. However, most of the benefit is already achieved with CoT prompting, leaving little room for further improvement through few-shot exemplars. For Claude-4.6, prompting strategy has only a marginal impact on explanation quality.

4.1.3 Cross-Granularity Consistency. While consistency across levels of granularity might be implicitly assumed, it is unclear whether token-level attributions reliably aggregate into meaningful attribute-level explanations. To investigate this aspect, we assess the consistency between attribute-level and token-level self-explanations. For feature attribution explanations, we compute the Kendall–Tau correlation between the ranking of attributes induced by their attribute-level scores and the ranking obtained by averaging the scores of their constituent tokens. A positive correlation indicates hierarchical consistency. For counterfactual explanations, we evaluate cross-level consistency by computing the cosine similarity between the sets of attribute values identified as counterfactuals at the attribute level and the corresponding sets derived from token-level counterfactual explanations.

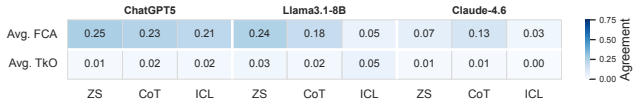
Figure 7 reports the results of this analysis. Figure 7(a) shows that attribute-level and token-level feature attribution explanations are often misaligned: correlations are often close to zero, indicating negligible agreement. Across models, the highest, albeit still modest, correlation values are observed for ChatGPT-5 and Claude-4.6-Sonnet. Figure 7(b) reports the cosine similarity scores for counterfactual explanations. In this case, a slightly stronger alignment between attribute-level and token-level explanations emerges. ChatGPT-5 achieves the largest alignment on average, while Claude-4.6-Sonnet shows alignment peaks for specific datasets (and prompts). However, across datasets, no consistent trend is observed.

4.1.4 Alignment of feature attribution and Counterfactual explanations. To quantify the consistency between feature attribution and counterfactual explanations, we introduce a measure that captures the fraction of the total attribution score concentrated on the features modified by a counterfactual.

Let x be an input record pair and let $\Delta(x)$ denote the set of features (attributes or tokens) modified by a counterfactual explanation for x. Let $s(f)$ be the attribution score assigned to feature f, and let $\mathcal{F}(x)$ denote the set of all features of x. We define the attribution score mass as $S(x) = \sum_{f \in \mathcal{F}(x)} s(f)$. We then define the *Feature attribution-Counterfactual Alignment* (FCA) for input x as: $\mathrm{FCA}(x) = \frac{\sum_{f \in \Delta(x)} s(f)}{S(x)}$. Higher values of $\mathrm{FCA}(x)$ indicate stronger

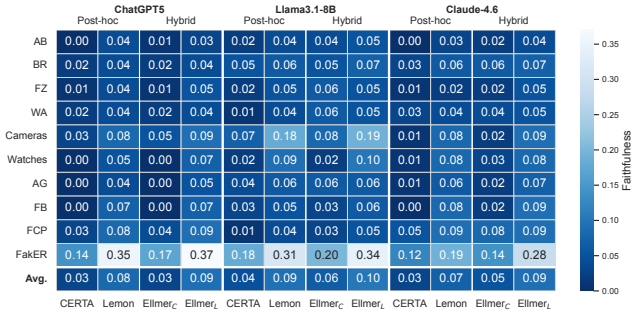


(a) Attribute-level (↑)

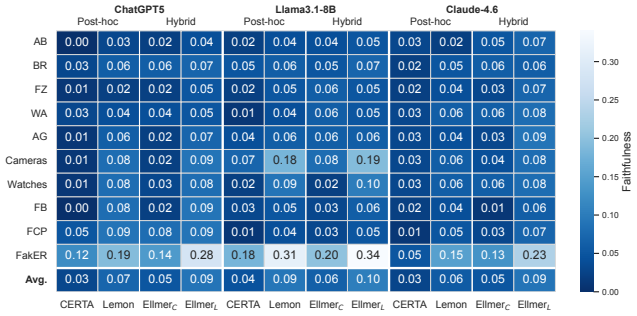


(b) Token-level (↑)

Figure 8: Feature attribution and counterfactual agreement of self-explanations (↑).



(a) Attribute-level (↓).



(b) Token-level (↓).

Figure 9: Faithfulness of feature attribution explanations generated by post-hoc and hybrid methods. Values close to 0 (darker) are better (↓). ELLMER$_C$ and ELLMER$_L$ refer to instantiating ELLMER with the post-hoc method CERTA and LEMON, respectively.

agreement between feature attribution and counterfactual explanations, as the features whose modification flips the prediction also account for a larger fraction of the model's feature attribution.

As a complementary measure, we also evaluate the overlap between counterfactual modifications and the most important features identified by feature attribution explanations. Let Top-$k(x) \subseteq \mathcal{F}(x)$ denote the set of the k features with the highest attribution scores. We define the *top-k overlap* as $\text{TkO}(x) = \frac{|\Delta(x) \cap \text{Top-}k(x)|}{|\Delta(x)|}$. This metric measures the fraction of features modified by the counterfactual that are also among the top-k most important features. Higher values indicate stronger alignment between feature attribution and counterfactual explanations, while lower values suggest that counterfactual changes affect features deemed less important by feature attribution explanations.

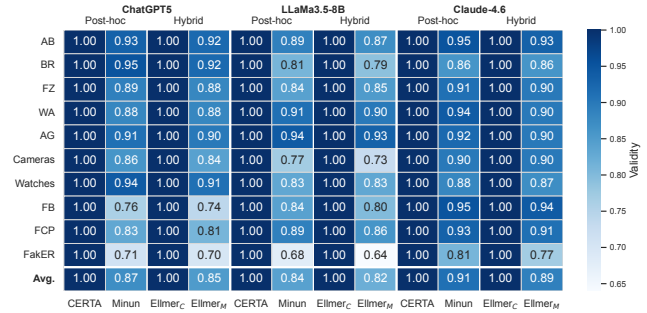
Figure 8 reports the average FCA and Top-k Overlap (TkO) aggregated across the considered datasets. For ChatGPT-5 and Llama-3.1 (especially), both metrics attain low values across prompting strategies, indicating limited agreement between feature attribution and counterfactual explanations. This suggests that, for those models, features receiving high attribution scores are not necessarily those whose modification is sufficient to alter the model's prediction. For Claude-4.6 we observe a higher degree of agreement between feature attribution and counterfactual explanations, across different prompts. At the attribute level, ICL improves alignment for Claude-4.6 and LLaMA 3.1–8B, leading to a higher concentration of attribution score on attributes modified by counterfactual explanations. In contrast, ChatGPT-5 achieves stronger alignment under ZS prompting, while CoT and ICL tend to degrade alignment, suggesting that explicit demonstrations or step-by-step reasoning may be less beneficial for stronger models. At the token level, alignment further decreases in most configurations, likely due to the substantially larger feature space and more diffuse attribution score distributions. No specific prompt strategy seems to outperform the others, token-level explanations remain weakly aligned with counterfactual evidence across all models. These results suggest that prompting strategies can influence the stability of feature attribution explanations but do not consistently improve their alignment with counterfactual explanations, particularly at finer levels of granularity. On the other hand, different models might inherently produce explanations with different degrees of alignment (but this won't necessarily translate into higher effectiveness, as reported in Figures 5 and 6).

4.2 RQ-2: Are post-hoc explanations more trustworthy?

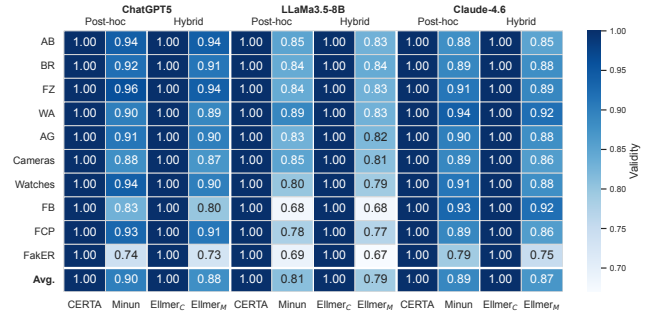
To answer this research question, analogously to the analysis for RQ-1, we assess trustworthiness by measuring faithfulness for feature attribution, and validity for counterfactual explanations.

4.2.1 Faithfulness and Validity of Self-Explanations. Figure 9 reports faithfulness scores for post-hoc and hybrid explanation methods. Compared to self-explanations (previously reported in Figure 5), post-hoc and hybrid approaches exhibit substantially higher trustworthiness, with faithfulness values consistently below 0.1 across datasets and models, at both the attribute and token levels. In contrast, self-explanations yield higher (hence worst) faithfulness values, ranging from 0.15 to 0.32 at the attribute level, and from 0.13 to 0.31 at the token level. Overall, the quality of the explanation is not influenced by the LLM, both at the attribute and token-level: for the same explanation method, the results provided by the three LLM are very close.

For counterfactual explanations, Figure 10 reports the results of our assessment, showing that also in this case post-hoc and hybrid approaches consistently outperform self-explanations (in Figure 6). At the attribute level, validity ranges from 0.82 to 1 for post-hoc and hybrid methods, compared to 0.63–0.81 for self-explanations.

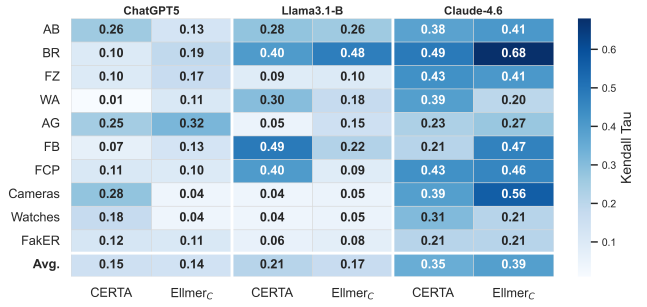


(a) Attribute-level (↑).

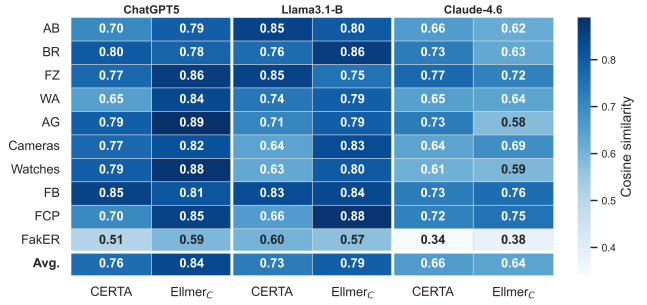


(b) Token-level (↑).

Figure 10: Validity of counterfactual explanations. Values close to 1 (darker) are better (↑). ELLMER$_C$ and ELLMER$_M$ refer to instantiating ELLMER with the post-hoc method CERTA and MINUN, respectively.



(a) Kendall-Tau correlation between attribute and token-level post-hoc and hybrid explanations (↑).



(b) Cosine similarity between attribute and token-level counterfactual post-hoc and hybrid explanations (↑).

Figure 11: Cross granularity consistency of post-hoc and hybrid explanations. Values close to 1 (darker) are better (↑).

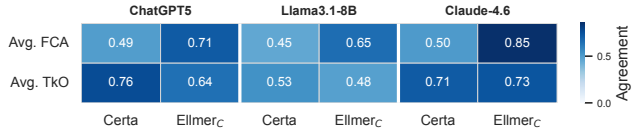
At the token level, validity ranges from 0.67 to 1 for post-hoc and hybrid methods, while self-explanations exhibit comparable but generally lower performance. Notably, validity is consistently equal to 1 across all datasets and models when using CERTA and ELLMER$_C$.

In contrast to self-explanations, post-hoc and hybrid explanations exhibit stable performance across datasets, with no marked degradation on token-dense records. This behavior suggests that post-hoc and hybrid approaches effectively decouple explanation quality from dataset complexity. The only exception is FakER, a stress-test dataset with extreme feature cardinality, where performance slightly degrades for feature attribution explanations, while counterfactual explanations remain comparable to other datasets.

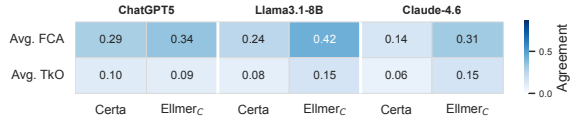
It is important to observe that the hybrid approaches, represented by the ELLMER variants, achieve trustworthiness levels comparable to their corresponding post-hoc methods. For feature attribution explanations (Figure 9), the performance of ELLMER$_C$ closely matches that of CERTA, with an average difference of 0.01 across models, while the results obtained by LEMON are comparable to those of the hybrid variant ELLMER$_L$. Similarly, for counterfactual explanations (Figure 10), ELLMER$_C$ and CERTA achieve identical performance, and MINUN and ELLMER$_M$ exhibit comparable validity scores.

For both faithfulness and validity, all the pairwise comparisons between self-explainers and post-hoc method are statistically significant ($p < 0.05$), with post-hoc methods consistently achieving lower faithfulness and higher validity. Mean differences in faithfulness span 0.048 (ChatGPT-5 CoT vs. LEMON, token level) to 0.247 (LLaMA ZS vs. CERTA, token level). Mean validity gaps range from −0.057 (Claude ICL vs. MINUN, attribute level) to −0.379 (LLaMA ZS vs. CERTA, token level), with CERTA consistently showing larger gaps relative to MINUN/LEMON.

4.2.2 Cross-Granularity Consistency. Figure 11 reports the cross-granularity consistency of post-hoc and hybrid explanations, measured by (a) the Kendall–Tau correlation between attribute-level and token-level feature attribution explanations, and (b) the cosine similarity between attribute-level and token-level counterfactual explanations. Due to space constraints, we only report consistency results for CERTA and ELLMER$_C$, which consistently achieved the strongest faithfulness and validity scores in preliminary experiments. Across all models, datasets, and explanation methods, Kendall–Tau values are consistently positive but modest, indicating weak yet systematic agreement rather than random noise. On average, CERTA achieves higher Kendall–Tau values than ELLMER$_C$ across all considered models; however, for ChatGPT-5 the difference is negligible, with the two methods exhibiting comparable average alignment. The degree of feature attribution alignment varies notably across models and datasets. In particular, post-hoc explanations with LLaMA 3.1–8B show stronger alignment on smaller product datasets, whereas ChatGPT-5 achieves higher alignment on datasets with larger and richer records, such as WDC and FakER. Overall, both post-hoc and hybrid explanations consistently improve cross-granularity feature attribution alignment compared to self-explanations (Figure 7), especially for smaller models. Turning to counterfactual explanations,



(a) Attribute-level (↑)



(b) Token-level (↑)

Figure 12: Feature attribution and counterfactual agreement of post-hoc and hybrid explanations. Values close to 1 (darker) are better (↑).

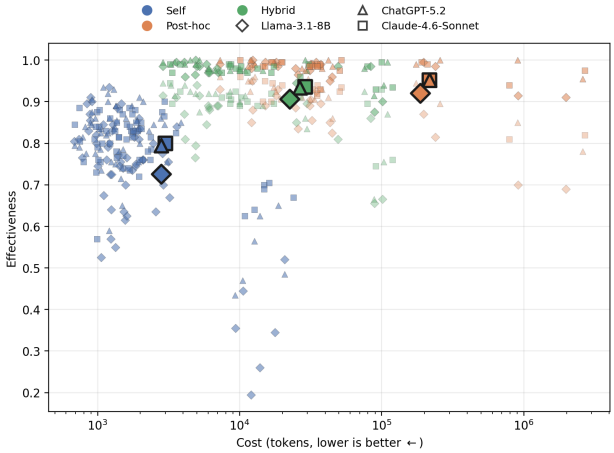


Figure 13: Effectiveness - Cost tradeoffs.

cosine similarity values remain above 0.63 across all models and datasets, with the exception of FakER, and average between 0.72 and 0.84. Hybrid explanations achieve slightly higher similarity than their post-hoc counterparts on average. The lower similarity observed for FakER is consistent with the substantially larger record size and token space of this dataset. With the exception of AG, counterfactual similarity values are markedly higher than those obtained for self-explanations, indicating a stronger and more stable cross-granularity alignment for post-hoc and hybrid methods. Overall, these results show that post-hoc and hybrid explanations substantially improve cross-granularity consistency compared to self-explanations. While feature attribution alignment remains modest, it is systematic and non-random, and counterfactual explanations exhibit a much stronger and more stable agreement across levels of granularity. This further supports the effectiveness of hybrid approaches in combining reliability and efficiency without compromising explanatory coherence.

4.2.3 Feature attribution–Counterfactual Alignment. Figure 12 reports the agreement between feature attribution and counterfactual explanations for post-hoc and hybrid approaches, measured through FCA and Top-k Overlap at both attribute and token levels. At the attribute level (Figure 12a), FCA values indicate that a substantial fraction of feature attribution scores is concentrated on attributes modified by counterfactual explanations. The hybrid approach Ellmer$_C$ consistently improves SCA over CERTA, particularly for ChatGPT-5. Similarly, Top-k Overlap reaches high values, showing that counterfactually modified attributes tend to appear among the ones with the highest attribution scores. These results contrast those obtained for self-explanations (Figure 8), where feature attribution and counterfactual explanations exhibited weak or negative alignment. This comparison highlights that post-hoc and hybrid methods induce a substantially more coherent relationship between attribution and counterfactual evidence. At the token level (Figure 12b), both SCA and Top-k Overlap are lower than at the attribute level, reflecting the larger feature space. These results indicate that post-hoc and hybrid approaches achieve a more reliable alignment between feature attribution and counterfactual explanations than self-explanations.

4.3 RQ-3: Is it feasible to compute post-hoc explanations for LLMs?

Generating post-hoc explanations requires multiple predictions on samples created by perturbing the features of the original input. This is especially costly for token-level explanations, which demand significantly more predictions, increasing computational overhead.

Figure 13 summarizes the trade-off between explanation effectiveness and cost (expressed as number of tokens consumed by the LLM) across all explainers, models, and datasets. Three regions emerge clearly. Self-explainers occupy the low-cost regime but exhibit substantially lower and more variable effectiveness. Post-hoc methods achieve the highest effectiveness, but at significantly higher token cost. Hybrid Ellmer variants lie between these extremes, matching the effectiveness of post-hoc methods while requiring one to two orders of magnitude fewer tokens. To validate these observations, we perform paired t-tests across all 60 experimental conditions. Both post-hoc and hybrid methods significantly outperform self-explainers in effectiveness (all $p < 10^{-20}$), while post-hoc methods retain a small but significant advantage over hybrid methods. The opposite ordering emerges for cost: self-explainers are the cheapest, post-hoc methods the most expensive, and hybrid methods occupy an intermediate position, with all pairwise differences being significant ($p < 10^{-21}$). Overall, the results identify a clear Pareto frontier. Self-explainers minimize cost at the expense of explanation quality, whereas post-hoc methods maximize effectiveness but incur substantial computational overhead. Hybrid Ellmer variants provide the most favourable trade-off, recovering near post-hoc trustworthiness at a fraction of the cost and consistently bridging the gap between reliability and scalability across models and granularities.

4.4 Correctness stratification

Figure 14 reports the mean difference in faithfulness and validity between correct and incorrect predictions (Δ = correct − incorrect) for each explainer and model. A negative Δ faithfulness indicates that explanations are more faithful when the underlying prediction is correct. A pronounced effect emerges for Claude-4.6 self-explanations. Under zs and COT, faithfulness is substantially higher for correct than incorrect predictions ($\Delta = -0.471$ and -0.480,

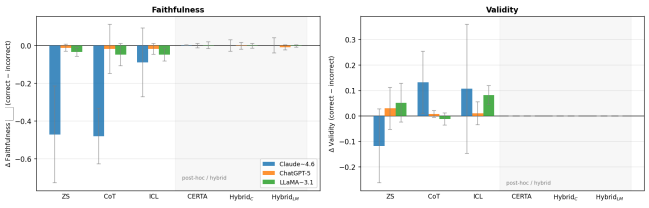


Figure 14: Mean difference in faithfulness and validity between correct and incorrect predictions.

respectively), whereas ChatGPT-5 and LLaMA-3.1 remain close to zero (all within $[-0.05, 0]$). ICL largely mitigates the effect for Claude, reducing the gap to -0.090, suggesting that few-shot exemplars help anchor explanations to the input regardless of prediction outcome. Validity shows a weaker pattern. Claude-4.6 exhibits moderate differences between correct and incorrect predictions (ZS: -0.117, COT: $+0.132$, ICL: $+0.107$), while ChatGPT-5 and LLaMA-3.1 again remain close to zero across prompting strategies. In contrast, post-hoc and hybrid methods (CERTA, ELLMERC, and ELLMERLM) consistently exhibit $\Delta \approx 0$ for both metrics across all models, indicating that explanation quality is largely independent of prediction correctness. This invariance represents a qualitative advantage over self-explanation approaches and is only partially mitigated through improved prompting.

4.5 Ellmer Sensitivity Analysis

Both variants of ELLMER identify the same operating point, with $\phi = 0.5$ providing the best quality–efficiency trade-off. ELLMER$_C$ is largely insensitive to both ϕ and model choice: validity remains close to 1.0, faithfulness stays within a narrow range, and token consumption increases only modestly as ϕ grows. In contrast, ELLMER$_{LM}$ is more sensitive to both factors. As ϕ increases from 0.25 to 0.5, faithfulness decreases while validity improves; beyond $\phi = 0.5$, both metrics largely plateau, providing little justification for better values despite the additional token cost. Overall, the sensitivity analysis indicates that $\phi = 0.5$ offers the most favorable balance between explanation quality and efficiency across both variants. Finally, since the parameter k (the initial number of features with the top attribution scores) is directly determined by ϕ, both parameters exhibit consistent behavior.

5 RELATED WORK

LLMs are currently employed across a range of data management tasks [15, 28]. Given their ability to contextualize the meaning of data, they can be considered an enabling technology especially in those tasks where the semantics of data plays a prominent role. [45] have shown the potential of LLMs for the ER task. On the other hand, despite their accuracy even in zero-shot and few-shot scenarios, they also exhibit a number of problematic behaviors, like not being able to respect the symmetric equality property (iff $A = B$ then $B = A$) as observed by [8]. The vulnerabilty of LLMs to adversarial prompts has been studied by [71].

Our work falls in the broad area of interpretable AI [16, 32] and causal inference [39]. Since understanding how LLMs inherently work is very hard [36, 50, 53, 69], research is focusing on the development of methods for interpreting and explaining LLMs, in different contexts and from different perspectives [27, 36, 69]. [50] reported instability in LLM self-explanations but did not address how to reconcile efficiency with explanation reliability, which is the goal of our hybrid ELLMER approach. [58] and [63] concentrated on CoT prompting, which provides improved performance along with implicit explanations by means of the illustrated step by step reasoning conducted by the LLM. However, [58] observed that such explanations are not always faithful. [64] studied gradient-based feature attributions and showed that CoT can improve the robustness of token-level attribution scores to input perturbations. [10] studied how to quantify the importance of portions of the prompt with respect to the output, as the length of the prompt varies.

Post-hoc explanations have been proven effective in a variety of tasks [20, 37]. As discussed in Section 3.2, several explanation systems have been proposed for the ER task [5, 6, 12, 13, 54, 60]. As we discussed along the paper, ELLMER is built on top of any post-hoc system based on perturbations and leverages the self-expanations of a LLM to overcome scalability issues.

6 CONCLUSIONS

This work empirically shows that LLM self-explanations for entity resolution are fluent and low-cost, yet exhibit notable trustworthiness limitations: relatively weak faithfulness, limited agreement between feature attribution and counterfactual explanations, and inconsistent cross-granularity behavior that prompting strategies alone do not fully address. Post-hoc methods offer substantially more reliable explanations, though at a considerable computational cost. ELLMER bridges this gap by using self-explanations to guide post-hoc exploration, preserving explanation quality while achieving meaningful reductions in runtime and token consumption.

An important open question left by this study concerns the *root causes* of self-explanation unreliability. The correctness stratification effect (reported in Section 4.4) suggests models might be doing post-hoc rationalization. That may indicate that harder instances induce both prediction errors and less coherent feature attribution simultaneously. Whether such behaviors reflect systematic artifacts in pretraining and instruction-tuning data [58] or a more fundamental property of autoregressive self-explainers remains an open empirical question. Investigating this through probing classifiers, causal tracing [38], or controlled experiments that vary training data composition would likely yield actionable insights for the broader community working on LLM interpretability in structured prediction tasks.

Future work might extend this evaluation to retrieval-augmented generation (RAG) setups, where dynamically retrieved context could further challenge explanation reliability, raising new questions about whether attributed features reflect the record pair, the retrieved evidence, or an uncontrolled mix of both.

ACKNOWLEDGMENTS

The work of Donatella Firmani has been partly funded by the APM project, MISE grant F-350020-01-X60, by the HORIZON Research and Innovation Action 101135576 INTEND "Intent-based data operation in the computing continuum" and the Sapienza Research Project "Knowledge Graph technologies for the next-generation Computing Continuum" (B83C2500088005).

REFERENCES

[1] A. Aberbach, M. Kejriwal, and K. Shen. Multipartite entity resolution: Motivating a k-tuple perspective (student abstract). *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(21):23434–23435, 2024.

[2] Anthropic. Claude 4.6 sonnet, 2026. Last accessed, April 30th 2026.

[3] P. Atanasova, J. G. Simonsen, C. Lioma, and I. Augenstein. A diagnostic study of explainability techniques for text classification. In B. Webber, T. Cohn, Y. He, and Y. Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 3256–3274. Association for Computational Linguistics, 2020.

[4] A. Baraldi, F. Del Buono, F. Guerra, M. Paganelli, M. Vincini, et al. An intrinsically interpretable entity matching system. In *Advances in Database Technology-EDBT*, volume 26, pages 645–657. OpenProceedings.org, 2023.

[5] A. Baraldi, F. Del Buono, M. Paganelli, and F. Guerra. Landmark explanation: An explainer for entity matching models. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 4680–4684, 2021.

[6] N. Barlaug. LEMON: explainable entity matching. *IEEE Trans. Knowl. Data Eng.*, 35(8):8171–8184, 2023.

[7] R. Benassi, F. Guerra, M. Paganelli, and D. Tiano. Explaining entity matching with clusters of words. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, pages 2325–2337. IEEE, 2024.

[8] L. Berglund, M. Tong, M. Kaufmann, M. Balesni, A. C. Stickland, T. Korbak, and O. Evans. The reversal curse: Llms trained on "a is b" fail to learn "b is a". *CoRR*, abs/2309.12288, 2023.

[9] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[10] O. Cífka and A. Liutkus. Black-box language model explanation by context length probing. In A. Rogers, J. L. Boyd-Graber, and N. Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 1067–1079. Association for Computational Linguistics, 2023.

[11] J. DeYoung, S. Jain, N. F. Rajani, E. Lehman, C. Xiong, R. Socher, and B. C. Wallace. Eraser: A benchmark to evaluate rationalized nlp models. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 4443–4458, 2020.

[12] V. di Cicco, D. Firmani, N. Koudas, P. Merialdo, and D. Srivastava. Interpreting deep learning models for entity resolution: an experience report using LIME. In R. Bordawekar and O. Shmueli, editors, *Proceedings of the Second International Workshop on Exploiting Artificial Intelligence Techniques for Data Management, aiDM@SIGMOD 2019, Amsterdam, The Netherlands, July 5, 2019*, pages 8:1–8:4. ACM, 2019.

[13] A. Ebaid, S. Thirumuruganathan, W. G. Aref, A. Elmagarmid, and M. Ouzzani. Explainer: entity resolution explanations. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, pages 2000–2003. IEEE, 2019.

[14] M. Fan, X. Han, J. Fan, C. Chai, N. Tang, G. Li, and X. Du. Cost-effective in-context learning for entity resolution: A design space exploration. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, pages 3696–3709. IEEE, 2024.

[15] J. Freire, G. Fan, B. Feuer, C. Koutras, Y. Liu, E. Peña, A. S. Santos, C. T. Silva, and E. Wu. Large language models for data discovery and integration: Challenges and opportunities. *IEEE Data Eng. Bull.*, 49(1):3–31, 2025.

[16] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42, 2018.

[17] Y. Guo, L. Chen, Z. Zhou, B. Zheng, Z. Fang, Z. Zhang, Y. Mao, and Y. Gao. Camper: An effective framework for privacy-aware deep entity resolution. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 626–637, 2023.

[18] S. Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

[19] A. Jacovi and Y. Goldberg. Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness? In *58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 4198–4205. Association for Computational Linguistics (ACL), 2020.

[20] S. Jesus, C. Belém, V. Balayan, J. Bento, P. Saleiro, P. Bizarro, and J. Gama. How can i choose an explainer? an application-grounded evaluation of post-hoc explanations. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 805–815, 2021.

[21] J. V. Jeyakumar, J. Noor, Y.-H. Cheng, L. Garcia, and M. Srivastava. How can i explain this to you? an empirical study of deep neural network explanation methods. *Advances in neural information processing systems*, 33:4211–4222, 2020.

[22] Z. Ji, X. Wang, Z. Luo, Z. Xie, and M. Zhang. Optimized batch prompting for cost-effective llms. *PVLDB*, 18(7):2172–2184, 2025.

[23] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[24] A.-H. Karimi, B. Schölkopf, and I. Valera. Algorithmic recourse: from counterfactual explanations to interventions. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 353–362, 2021.

[25] M. G. Kendall. Rank correlation methods. 1948.

[26] R. Kommiya Mothilal, D. Mahajan, C. Tan, and A. Sharma. Towards unifying feature attribution and counterfactual explanations: Different means to the same end. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '21, page 652–663, New York, NY, USA, 2021. Association for Computing Machinery.

[27] S. Krishna, J. Ma, D. Slack, A. Ghandeharioun, S. Singh, and H. Lakkaraju. Post hoc explanations of language models can improve language models. *Advances in Neural Information Processing Systems*, 36, 2024.

[28] G. Li, X. Zhou, and X. Zhao. Llm for data management. *PVLDB*, 17(12):4213–4216, 2024.

[29] H. Li, S. Li, F. Hao, C. J. Zhang, Y. Song, and L. Chen. Booster: Leveraging large language models for enhancing entity resolution. In *Companion Proceedings of the ACM on Web Conference 2024*, pages 1043–1046, 2024.

[30] Y. Li. A practical survey on zero-shot prompt design for in-context learning. In R. Mitkov and G. Angelova, editors, *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 641–647, Varna, Bulgaria, Sept. 2023. INCOMA Ltd., Shoumen, Bulgaria.

[31] Y. Li, J. Li, Y. Suhara, A. Doan, and W.-C. Tan. Deep entity matching with pre-trained language models. *PVLDB*, 14(1):50–60, 2020.

[32] Z. C. Lipton. The mythos of model interpretability. *Commun. ACM*, 61(10):36–43, Sept. 2018.

[33] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

[34] Y. Lu, M. Bartolo, A. Moore, S. Riedel, and P. Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In S. Muresan, P. Nakov, and A. Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[35] A. Madsen, S. Chandar, and S. Reddy. Are self-explanations from large language models faithful? In L.-W. Ku, A. Martins, and V. Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 295–337, Bangkok, Thailand, Aug. 2024. Association for Computational Linguistics.

[36] A. Madsen, S. Reddy, and S. Chandar. Post-hoc interpretability for neural nlp: A survey. *ACM Comput. Surv.*, 55(8), Dec. 2022.

[37] A. Madsen, S. Reddy, and S. Chandar. Post-hoc interpretability for neural nlp: A survey. *ACM Computing Surveys*, 55(8):1–42, 2022.

[38] K. Meng, D. Bau, A. Andonian, and Y. Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

[39] R. Moraffah, M. Karami, R. Guo, A. Raglin, and H. Liu. Causal interpretability for machine learning-problems, methods and evaluation. *ACM SIGKDD Explorations Newsletter*, 22(1):18–33, 2020.

[40] R. K. Mothilal, A. Sharma, and C. Tan. Explaining machine learning classifiers through diverse counterfactual explanations. In M. Hildebrandt, C. Castillo, L. E. Celis, S. Ruggieri, L. Taylor, and G. Zanfir-Fortuna, editors, *FAT* '20: Conference on Fairness, Accountability, and Transparency, Barcelona, Spain, January 27-30, 2020*, pages 607–617. ACM, 2020.

[41] S. Mudgal, H. Li, T. Rekatsinas, A. Doan, Y. Park, G. Krishnan, R. Deep, E. Arcaute, and V. Raghavendra. Deep learning for entity matching: A design space exploration. In *Proceedings of the 2018 International Conference on Management of Data*, pages 19–34, 2018.

[42] N. Nananukul, K. Sisaengsuwanchai, and M. Kejriwal. Cost-efficient prompt engineering for unsupervised entity resolution in the product matching domain. *Discover Artificial Intelligence*, 4(1):56, 2024.

[43] M. Paganelli, D. Tiano, and F. Guerra. A multi-facet analysis of bert-based entity matching models. *The VLDB Journal*, 33(4):1039–1064, 2024.

[44] R. Peeters and C. Bizer. Dual-objective fine-tuning of bert for entity matching. *PVLDB*, 14(10):1913–1921, jun 2021.

[45] R. Peeters and C. Bizer. Using chatgpt for entity matching. In A. Abelló, P. Vassiliadis, O. Romero, R. Wrembel, F. Bugiotti, J. Gamper, G. Vargas-Solar, and E. Zumpano, editors, *New Trends in Database and Information Systems - ADBIS 2023 Short Papers, Doctoral Consortium and Workshops: AIDMA, DOING, K-Gals, MADEISD, PeRS, Barcelona, Spain, September 4-7, 2023, Proceedings*, volume 1850 of *Communications in Computer and Information Science*, pages 221–230. Springer, 2023.

[46] A. Primpeli, R. Peeters, and C. Bizer. The wdc training dataset and gold standard for large-scale product matching. In *Companion Proceedings of the 2019 World Wide Web Conference (WWW '19 Companion)*, pages 381–386, 2019.

[47] M. T. Ribeiro, S. Singh, and C. Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[48] Y. Rong, T. Leemann, T.-T. Nguyen, L. Fiedler, P. Qian, V. Unhelkar, T. Seidel, G. Kasneci, and E. Kasneci. Towards human-centered explainable ai: A survey of user studies for model explanations. *IEEE transactions on pattern analysis and machine intelligence*, 2023.

[49] W. Samek, A. Binder, G. Montavon, S. Lapuschkin, and K.-R. Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673, 2016.

[50] H. Shiyuan, M. Siddarth, J. Shreedhar, Z. Yilun, and H. G. Leilani. Can large language models explain themselves? a study of llm-generated self-explanations. *CoRR*, abs/2310.11207, 2023.

[51] K. Sisaengsuwanchai, N. Nananukul, and M. Kejriwal. How does prompt engineering affect chatgpt performance on unsupervised entity resolution? *arXiv preprint arXiv:2310.06174*, 2023.

[52] D. Slack, A. Hilgard, S. Singh, and H. Lakkaraju. Reliable post hoc explanations: Modeling uncertainty in explainability. *Advances in neural information processing systems*, 34:9391–9404, 2021.

[53] A. Srivastava, A. Rastogi, A. Rao, A. A. M. Shoeb, A. Abid, A. Fisch, A. R. Brown, A. Santoro, A. Gupta, A. Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on machine learning research*, 2023.

[54] T. Teofili, D. Firmani, N. Koudas, V. Martello, P. Merialdo, and D. Srivastava. Effective explanations for entity resolution models. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 2709–2721. IEEE, 2022.

[55] T. Teofili, D. Firmani, N. Koudas, P. Merialdo, and D. Srivastava. Certem: explaining and debugging black-box entity resolution systems with certa. *PVLDB*, 15(12):3642–3645, 2022.

[56] S. Thirumuruganathan, M. Ouzzani, and N. Tang. Explaining entity resolution predictions: Where are we and what needs to be done? In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics*, pages 1–6, 2019.

[57] S. Thirumuruganathan, M. Ouzzani, and N. Tang. Explaining entity resolution predictions: Where are we and what needs to be done? In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics*, HILDA '19, New York, NY, USA, 2019. Association for Computing Machinery.

[58] M. Turpin, J. Michael, E. Perez, and S. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36:74952–74965, 2023.

[59] S. Wachter, B. Mittelstadt, and C. Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

[60] J. Wang and Y. Li. Minun: evaluating counterfactual explanations for entity matching. In M. Boehm, P. Varma, and D. Xin, editors, *DEEM '22: Proceedings of the Sixth Workshop on Data Management for End-To-End Machine Learning Philadelphia, PA, USA, 12 June 2022*, pages 7:1–7:11. ACM, 2022.

[61] T. Wang, X. Chen, H. Lin, X. Chen, X. Han, L. Sun, H. Wang, and Z. Zeng. Match, compare, or select? an investigation of large language models for entity matching. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 96–109, 2025.

[62] X. Wang and M. Yin. Are explanations helpful? a comparative study of the effects of explanations in ai-assisted decision-making. In *Proceedings of the 26th International Conference on Intelligent User Interfaces*, pages 318–328, 2021.

[63] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[64] S. Wu, E. M. Shen, C. Badrinath, J. Ma, and H. Lakkaraju. Analyzing chain-of-thought prompting in large language models via gradient-based feature attributions. *CoRR*, abs/2307.13339, 2023.

[65] Y. Xia, J. Chen, X. Li, and J. Gao. Aprompt4em: Augmented prompt tuning for generalized entity matching. *arXiv preprint arXiv:2405.04820*, 2024.

[66] A. Zeakis, G. Papadakis, D. Skoutas, and M. Koubarakis. Pre-trained embeddings for entity resolution: an experimental analysis. *PVLDB*, 16(9):2225–2238, 2023.

[67] A. Zeakis, G. Papadakis, D. Skoutas, and M. Koubarakis. An in-depth analysis of pre-trained embeddings for entity resolution. *The VLDB Journal*, 34(1):1–27, 2025.

[68] L. Zecchini, G. Simonini, S. Bergamaschi, and F. Naumann. Brewer: Entity resolution on-demand. *PVLDB*, 16(12):4026–4029, 2023.

[69] H. Zhao, H. Chen, F. Yang, N. Liu, H. Deng, H. Cai, S. Wang, D. Yin, and M. Du. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 2023.

[70] Z. Zhao, E. Wallace, S. Feng, D. Klein, and S. Singh. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pages 12697–12706. Pmlr, 2021.

[71] K. Zhu, J. Wang, J. Zhou, Z. Wang, H. Chen, Y. Wang, L. Yang, W. Ye, N. Z. Gong, Y. Zhang, and X. Xie. Promptbench: Towards evaluating the robustness of large language models on adversarial prompts. *CoRR*, abs/2306.04528, 2023.

[72] J. Zhuo, S. Zhang, X. Fang, H. Duan, D. Lin, and K. Chen. Prosa: Assessing and understanding the prompt sensitivity of llms. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1950–1976, 2024.