
# CAS 764- Advance Topics in Data Management

**Submitted by**

**Name:** **Arpita Bhattacharijee**

**ID:** [redacted]

**Date of Submission: 12th December 2025**

**Table of Contents**



## List of Tables



### List of Figures

# Ensuring Data Quality: An LLM-Enhanced Framework for Data Diagnosis and Resolution


## Abstract

Large language model (LLM), centric data quality assessment is a highly non-trivial task due to the heterogeneity, sparsity, and noise that characterize real-world tabular and semi-structured data. Core data quality operations such as schema matching, entity resolution, and data imputation require both semantic understanding and robust decision-making under uncertainty capabilities that traditional rule-based or purely statistical methods often fail to provide. In this work, we propose a unified, LLM centric framework that leverages LLM reasoning and semantic embeddings to address these data quality challenges in an integrated manner. Specifically, we investigate (i) schema matching using zero-shot and fine-tuned LLMs under varying attribute difficulty levels, (ii) entity resolution using embedding-based similarity augmented with lexical baselines for comparison, and (iii) entity-aware data imputation formulated as a prompt-based decision task. Our experimental results show that fine-tuned LLMs achieve substantial gains in schema matching accuracy compared to zero-shot settings, while SBERT-based entity resolution significantly outperforms TF-IDF and Jaccard similarity baselines. The proposed framework demonstrates clear advantages in capturing semantic relationships and handling heterogeneous representations across data sources. However, our findings also reveal that data imputation remains particularly challenging, with performance constrained by limited ground-truth availability and sparse entity-level evidence. This study provides a systematic evaluation of LLM-assisted data quality pipelines, demonstrating their promise while offering a balanced discussion of current limitations and directions for future research.


# 1 Introduction

Data quality diagnosis constitutes a core task in advance database management. To explain further , data quality diagnosis is the early step of data preparation for various task such as preparing data for predictive modelling, data analytics, insights generation and so on. Tabular data preparation encompasses the complete process of transforming raw, heterogeneous tables into coherent and analytically usable structures. Its significance is well established, as the reliability of downstream models is directly constrained by the quality of the underlying data, a relationship consistently highlighted through the “garbage in, garbage out” problem [1]. The task remains highly labour-intensive and dependent on expert decision-making, with empirical studies indicating that data scientists devote a substantial portion of their workload to data cleaning and organization[2].

Prior work highlights that, one of the key steps in data quality assessing: human-driven schema matching that depends heavily on expert intervention, which is costly and difficult to sustain in large deployments [3]. The manual nature of the task also introduces vulnerability to mistakes and inconsistencies, as cognitive biases and fatigue can influence decision-making [4]. In addition to that, Early work on tabular data preparation predominantly employed rule-based techniques and conventional learning models, including tree-based methods and shallow

neural networks[5]. For example, machine learning techniques have been applied extensively across a range of data-management tasks, including data cleaning [6], data analytics [7], query rewriting [8], and database diagnosis [9]. Despite their widespread use, conventional machine learning models face persistent challenges related to generalization and contextual inference. However, these systems were limited in their ability to capture the implicit structures and relationships embedded within tables, making them difficult to extend to more complex preparation tasks. Additionally, these methods frequently failed to represent fine-grained semantic dependencies essential for accurate processing, and their reliance on task-specific designs with rigid assumptions imposed significant constraints on generalization. As a result, these persistent demands have driven ongoing efforts to develop an automated and a scalable framework for supporting effective preparation of structured data.

Leveraging large language models(llm)[10] for data-quality diagnosis offers a wide possibility to overcome the limited contextual understanding found in traditional data cleaning or machine-learning approaches. Large language models have recently demonstrated remarkable progress on tasks that demand strong semantic understanding [3]. LLM exhibits an impressive capability to transfer their knowledge to new tasks without requiring task-specific fine-tuning, extending even to domains far removed from their original training scope, including a range of data-centric applications [11,12]. Developing a robust LLM-enhanced data management system is critical for effectively integrating large language models into real-world data workflows. This work addresses three key challenges. First, heterogeneous data sources that includes structured tables and unstructured or semi-structured documents must be systematically exploited to ground model outputs and reduce hallucination. Such examples are schema-aware processing and representation-based retrieval rather than direct free-form generation. Second, the computational and financial cost of invoking LLMs remains high; therefore, it is essential to minimize unnecessary LLM usage by leveraging lightweight components such as embedding models, similarity search, and deterministic decision rules, reserving LLMs only for ambiguous or high-value cases. Third, complex data management tasks typically consist of multiple interdependent operations (e.g., schema alignment, entity resolution, and record consolidation). Efficiently orchestrating these operations into modular, reusable pipelines is necessary to improve both execution efficiency and overall system effectiveness. This vision emphasizes a hybrid architecture in which LLMs augment rather than replace core data management primitives.

In this paper, we study a multi-stage LLM-enhanced data integration pipeline that combines schema matching, entity resolution, and data imputation. We first investigate schema matching using a fine-tuned large language model, where schema documentation and attribute descriptions are leveraged to align heterogeneous schemas. Rather than relying solely on zero-shot prompting, the model is adapted to the domain to improve robustness and reduce ambiguity in schema alignment. Building on the matched schemas, we perform entity resolution using a hybrid approach that integrates rule-based attribute extraction with embedding-based similarity. Specifically, records are encoded using a sentence-level embedding model that maps textual attribute representations into a continuous vector space, where semantically related entities are positioned closer to one another. This embedding space captures contextual relationships between attributes such as shared brand names, model identifiers, and descriptive patterns beyond exact lexical overlap, enabling robust candidate

retrieval even in the presence of noisy, incomplete, or heterogeneous records. Finally, we study data imputation as a downstream task that explicitly depends on entity resolution, where prompt-engineered LLM queries are used to infer missing values from consolidated entity representations rather than raw records. This design allows LLMs to be invoked selectively and contextually, reducing unnecessary overhead while preserving data consistency across tasks. Based on this vision , this research will aim to address these questions:

I. How does LLM-based schema matching perform in zero-shot and fine-tuned settings compared to string-similarity-based methods, and how does it compare with string-similarity-based methods?
II. How do embedding-based approaches compare with lexical similarity methods for entity resolution across heterogeneous data sources?
III. How does the quality of entity resolution affect downstream data imputation, and how do traditional method compare with LLM-based imputation when using resolved entities?

To address these research questions, we introduce an LLM-centric data diagnosis framework in Section 3 (Methodology). In section 4 (Experiments), describes the adopted methods, task scope, experimental settings, and the end-to-end pipelines for schema matching, entity resolution, and data imputation. The research questions are evaluated in Section 5 (Results), where we present a detailed analysis of model performance and discuss the findings for each task. Finally, Section 6 concludes the paper by summarizing the robustness of the proposed framework and outlining directions for future research.

# 2 Background and Related Work

Large language models(llm) learn patterns of human language and can produce coherent text, supporting both task-specific fine-tuning and broad reuse in zero-shot settings across many applications [13, 14, 15, 16]. Alongside them, embedding models such as BERT and Ada [17] map text into rich contextual vectors that have driven notable improvements across many NLP benchmarks [18, 19]. These vector representations also make it possible to retrieve relevant information using semantic similarity rather than relying on exact word matches [17, 20]. Recent studies have explored the use of LLMs in core data-preprocessing activities including detecting errors, filling missing values, resolving entities, and aligning schemas [11,15] with results that indicate strong potential. However, these efforts still encounter constraints related to high computational cost, limited efficiency, and insufficient testing on datasets that capture the full complexity of real-world schema-matching problems.

## 2.1 Schema Matching

Schema matching focuses on learning a mapping function that aligns attributes from a source schema to their corresponding attributes in a target schema. A common recent approach is to use prompt-based LLM reranking, where an initial set of candidate matches is first generated and then refined using LLMs through carefully designed prompts[21]. Existing methods differ mainly in two aspects: how candidate matches are generated and how prompts are constructed.

For candidate generation, ReMatch transforms target schema elements into retrievable documents and applies semantic similarity to identify potential matches[3]. KG-RAG4SM incorporates external knowledge graphs by encoding graph-structured information and external knowledge to support semantic alignment [22]. Prompt-Matcher relies on traditional schema-matching techniques to produce probabilistic candidate sets before LLM refinement[23], while Magneto uses a lightweight small language model to retrieve candidates efficiently [24].In addition to candidate generation, prior work has explored diverse prompt design strategies to improve LLM-based reranking. Matchmaker iteratively updates candidate matches using confidence scores and multi-round feedback to progressively improve alignment quality [25].

Overall, prompt-based LLM reranking approaches benefit from operating on small schema sizes and pre-filtered candidate sets, which enables efficient refinement and higher precision through richer contextual reasoning. However, these methods are sensitive to the quality of initial candidate generation; insufficient recall at this stage can result in valid schema matches being overlooked.

## 2.2 Entity Matching

Entity matching aims to determine whether two records drawn from different datasets refer to the same real-world entity. Following the widely adopted block-and-match framework [26], recent work has explored language models as matchers through both prompt engineering and fine-tuning strategies.

Several studies investigate the use of prompting to perform entity matching without additional training. This study [27] evaluates different prompt formulations and demonstrate that even simple zero-shot prompts can achieve strong performance. BATCHER [28] focuses on improving cost efficiency by batching multiple entity pairs and selecting informative in-context examples, significantly reducing token consumption while maintaining comparable accuracy. BoostER [29] introduces selective verification by first generating candidate matches with associated probabilities and then refining decisions through Bayesian adjustment. Fine-tuning has also been widely applied to entity matching, particularly using smaller language models. JointBERT [30] fine-tunes BERT jointly for binary matching and multi-class entity classification, enabling the model to exploit richer supervision signals. Ditto[31] further improves performance by augmenting pre-trained transformer models such as BERT and RoBERTa with task-specific textual transformations and domain knowledge. More recent work by [32] fine-tunes both SLMs and LLMs, showing that well-tuned smaller models consistently yield performance gains, while results for fine-tuned LLMs tend to be more variable.

Prompt-based entity matching offers strong flexibility, allowing models to adapt rapidly across domains without retraining. However, matching decisions can be unstable when prompts fail to capture subtle distinctions between ambiguous entities. This limitation motivates the need for lighter, more effective alternatives. Embedding-based methods offer a practical path forward: they capture semantic relationships efficiently, run at low computational cost, and avoid repeated LLM invocations. By pairing embeddings with similarity search and simple decision rules, the system can handle most routine data-quality tasks such as matching, filtering, and coarse classification without relying on expensive model calls. This creates space for an architecture where the LLM steps in only when deeper reasoning is necessary, reducing overall cost while maintaining strong performance.

## 2.3 Data Imputation

Data imputation aims to infer and populate missing or unobserved values in datasets. Similar to data repair, imputation is inherently a generative task, which has motivated the exploration of both LLM-centric approaches and LM-in-the-loop frameworks.

Several methods rely directly on large language models to perform imputation through prompting or fine-tuning. Study in this paper [11], demonstrate that LLMs can be prompted with few-shot examples to generate missing values without additional model training. Extending this idea, LLM Forest [33] constructs a bipartite graph for each feature and aggregates the outputs of multiple LLM-based few-shot learning components into a forest structure, using weighted voting to determine final imputation results. Fine-tuning strategies have also been investigated: the proposed study in this paper [34] applied LoRA-based fine-tuning to LLMs, incorporate existing dataset values as contextual knowledge within prompts, and generate candidate imputations accordingly. UnIMP [35] first constructs cell-level hypergraphs by using an LLM as an encoder to generate feature representations. High-order message passing is then applied to propagate information across the dataset, after which the LLM is used as a decoder to generate imputed values based on the aggregated representations.

LLM-driven techniques can infer missing values and adjust to varied data patterns with ease, yet they sometimes rely too heavily on surface correlations or generate answers that appear reasonable but are incorrect. Lightweight, LM-in-the-loop strategies address part of this problem by pairing LLM reasoning with structured components such as embedding models, similarity search, or rule-based checks. This hybrid setup offers more reliable and interpretable outcomes while keeping LLM usage selective. However, this design introduces additional coordination demands, as the system must effectively integrate rapid, deterministic components with the more complex reasoning capabilities of the LLM.

# 3 Methodology

As illustrated in Figure 1, We have proposed an LLM-centric framework for structured data quality assessment and resolution that addresses schema matching, entity resolution, and data imputation in a unified pipeline. The framework leverages instruction-tuned large language models, lightweight fine-tuning, and embedding-based retrieval to robustly handle heterogeneous, noisy, and incomplete product specification data. Each component is designed to operate independently while sharing standardized representations to ensure consistency across stages.

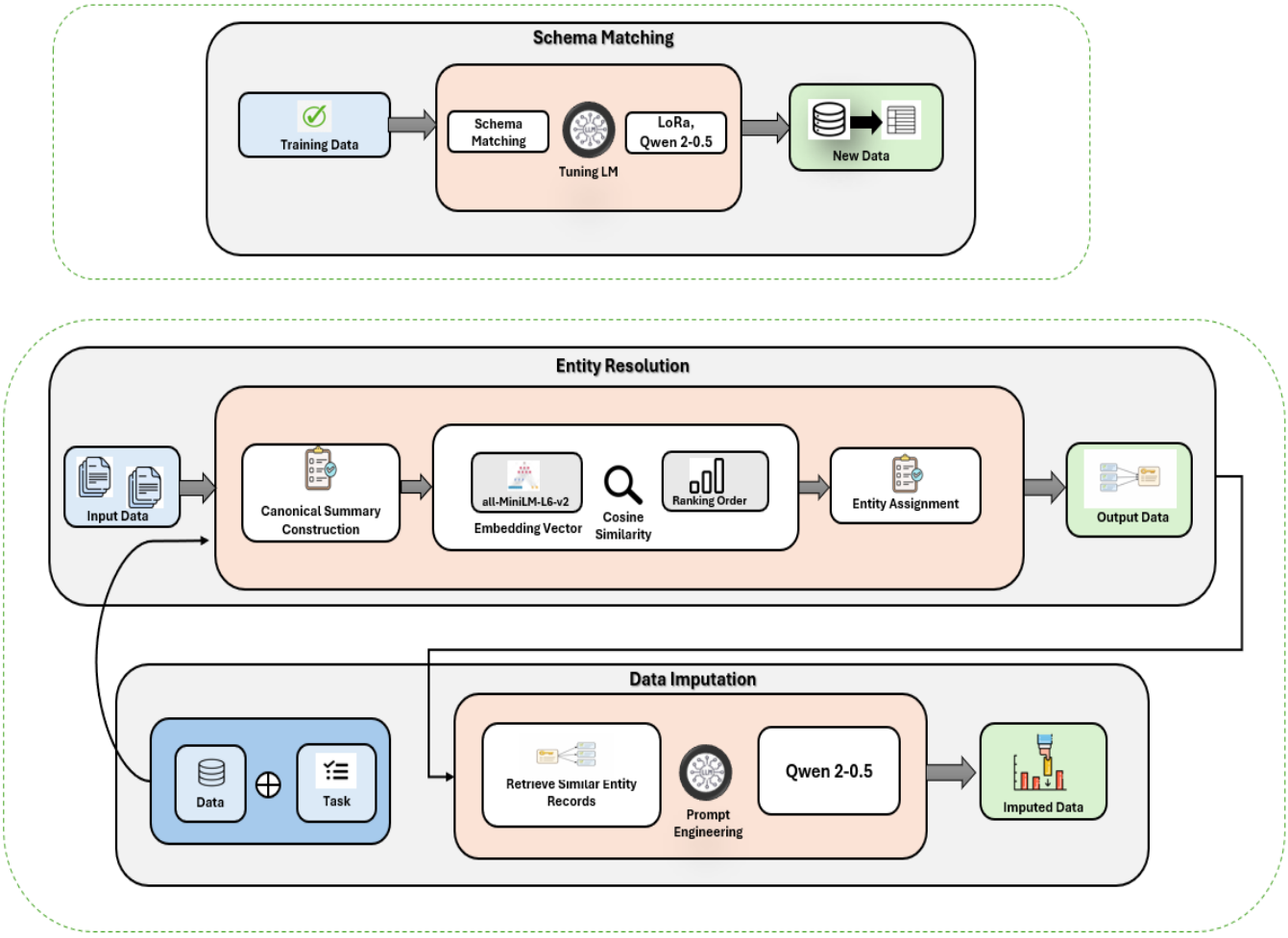



**Figure 1: Proposed LLM Enhanced Framework for Data Diagnosis and Resolution**

## 3.1 Schema Matching

The first task addressed in this research is schema matching, which aims to resolve heterogeneous and source-specific attribute names into a unified canonical schema. We formulate schema matching as a supervised canonical attribute classification problem, where the ground-truth dataset consists of raw attribute descriptions and their corresponding target attributes. As illustrated in Table 1, each record originates from a specific data source and includes a noisy attribute name along with its manually curated canonical resolution. Based on these records, we construct a schema-mapping dataset by extracting the source website, the

problematic attribute name, and its associated attribute value, and assigning the corresponding target attribute as the canonical property, as shown in Figure 2. The inclusion of attribute values is intentional, as values often provide additional semantic context that is not evident from attribute names alone.

To perform schema matching, we design an instruction-based prompt that explicitly frames the task as canonical property identification and constrains the model's output to a predefined list of canonical attributes. The prompt requires the model to select exactly one canonical property and to return only the property name, enabling deterministic outputs and straightforward evaluation. Using this prompt structure, we fine-tune a Qwen-2.5 language model with Low-Rank Adaptation (LoRA), which allows efficient task-specific adaptation while keeping the base model parameters frozen. This approach enables the model to learn robust mappings from heterogeneous, site-specific attribute expressions including synonyms, paraphrases, and noisy formulations to a unified canonical schema. During inference, the same prompt template is applied to unseen data, ensuring consistency between training and deployment and enabling a fair comparison with a zero-shot baseline that uses identical instructions but different model weights.

**Table 1: Defined Ground Truth Schema Data**

| Source attribute id | Target_attribute_name |
|---|---|
| www.alibaba.com//auto focus | auto_focus_automatic |
| www.eglobalcentral.co.uk//focus | auto_focus_automatic |
| www.flipkart.com//focus | auto_focus_automatic |
| buy.net//width | Camera_width |
| www.alibaba.com//effective pixels | Image_resolution |
| www.cambuy.com.au//exposure control | Exposure_compensation |
| www.canon-europe.com//photo effects | Picture_styles |

| site | record_id | current_attr | value | target_attr |
|---|---|---|---|---|
| buy.net | 4233 | camera type | Mirrorless Interchange | camera_type |
| buy.net | 4233 | depth | 1.8 in | camera_depth |
| buy.net | 4233 | effective megapi | 16100000 pixels | image_resolution |
| buy.net | 4233 | image sensor | CMOS | sensor_type |
| buy.net | 4233 | lcd screen size | 3 in | screen_size |
| buy.net | 4233 | lens mount | Micro Four Thirds | lens_mount_type |
| buy.net | 4233 | maximum video | 1920 x 1080 | video_resolution |
| buy.net | 4233 | memory card su | Secure Digital High Cap | external_memory_type |
| buy.net | 4233 | total pixels | 17200000 pixels | image_resolution |
| buy.net | 4233 | warranty inform | 1 year(s) | warranty_duration |
| buy.net | 4233 | width | 4.7 in | camera_width |
| buy.net | 4236 | brand name | Polaroid | brand |
| buy.net | 4236 | exposure control | Auto | exposure_mode |
| buy.net | 4239 | camera type | Compact Camera | camera_type |
| buy.net | 4239 | depth | 1 in | camera_depth |
| buy.net | 4239 | digital zoom | 4x | zoom_digital |
| buy.net | 4239 | effective megapi | 12.1 Megapixel | image_resolution |
| buy.net | 4239 | image sensor | BSI CMOS | sensor_type |
| buy.net | 4239 | lcd screen size | 3.2 in | screen_size |
| buy.net | 4239 | maximum video | 1920 x 1080 | video_resolution |
| buy.net | 4239 | memory card su | Secure Digital (SD) Carc | external_memory_type |
| buy.net | 4239 | optical zoom | 4.4 X | zoom_optical |
| buy.net | 4239 | width | 4 in | camera_width |
| buy.net | 4247 | battery model su | LP-E6N / LP-E6 | battery_type |
| buy.net | 4247 | brand name | Canon | brand |
| buy.net | 4247 | exposure control | Program AE Aperture Pr | exposure_mode |
| buy.net | 4247 | iso sensitivity | ISO 51200 | iso_sensitivity_max |
| buy.net | 4247 | longest shutter s | 30 Second | shutter_speed_min |

**Figure 2: Processed Training Data for Schema Matching**

## 3.2 Entity Resolution

As shown in Figure 1, the proposed entity resolution module aims to determine whether an incoming record refers to an existing real-world entity or represents a previously unseen entity. The method follows a representation retrieval decision paradigm and functions without supervised re-training at inference time.

Each known entity is represented through a canonical summary constructed from its associated records. Instead of relying on exact attribute names, the method identifies attribute values using substring-based matching over lowercased keys, allowing semantically equivalent fields (e.g., model, product title, SKU) to be mapped to a common attribute space. Using this mechanism, a fixed set of core attributes (as summarized in appendix) is extracted for each record. These attributes are then composed into a compact canonical textual signature by prioritizing identity-defining information (brand, model, and camera type) and appending discriminative specifications such as sensor characteristics, mount, lens information, and video capability in a structured format.

As define the pseudocode (Table 2), After an incoming record $R$ being transformed into a canonical representation, core attributes are extracted from the raw record and composed into a canonical textual signature $T_r$, using the predefined attributes. This signature captures the identity-relevant information of the record in a normalized and source-agnostic form. The textual signature is then encoded into a normalized embedding vector $v_r$, ensuring comparability with existing canonical entity representations.

**Table 2: Pseudocode for Entity Matching**

```
Input:
  R          /incoming record (raw data)
  C = {c1…cn}   / canonical entity signatures
  V = {v1…vn}   /embeddings of canonical signatures
  θ          / similarity threshold

Output:
  entity_id OR NEW_ENTITY

1. Record abstraction
   A_r ← extract_attributes(R)
   T_r ← build_canonical_signature(A_r)
   v_r ← embed_and_normalize(T_r)

2. Similarity scoring
   for each entity i in C:
      s_i ← cosine_similarity(v_r, v_i)

3. Candidate selection
   P ← top_K entities by s_i

4. Best match decision
   i* ← argmax(s_i) over P
   if s_{i*} ≥ θ:
      return entity_id(i*)
   else:
      return NEW_ENTITY
```

Next, semantic similarity scores are computed between the record embedding $v_r$ and the embeddings of all canonical entity signatures $\{v_1, \dots, v_n\}$ using cosine similarity. To reduce noise and improve efficiency, only the top-$K$ entities with the highest similarity scores are retained as candidate matches. Finally, the candidate with the highest similarity score is selected. If this score exceeds a predefined threshold $\theta$, the record is assigned to the corresponding entity. Otherwise, the record is classified as a new entity. This threshold-based decision mechanism enables open-world entity resolution by avoiding forced matches when no sufficiently similar entity exists.

## 3.3 Data Imputation

In the proposed framework, data imputation is performed as a post–entity-resolution step using prompt-based reasoning. For a record with missing attributes, the system first identifies its associated entity and retrieves all available records belonging to the same entity. These records are treated as evidence and are summarized to extract candidate values for the missing attribute along with their occurrence patterns. The summarized evidence, together with minimal contextual information from the target record such as product name or brand, is then provided to a selected large language model through a carefully designed prompt. Prompt engineering is used to frame imputation as a constrained decision task, instructing the model to select the most plausible value from the observed candidates rather than generating arbitrary content. The model returns both an imputed value and a confidence score, and predictions below a predefined confidence threshold are discarded. The hypothesis of this approach to enable accurate imputation by leveraging entity-level consistency and controlled LLM reasoning while avoiding uncontrolled generation.

# 4 Experiments

This section outlines the selected datasets, application domain, model configurations, and the experimental evaluations used to assess the effectiveness and efficiency of the proposed LLM-centric data diagnosis framework.

## 4.1 Dataset Description

For our experiments, we use the camera vertical of the Alaska benchmark[36] , which provides real-world product specifications collected from heterogeneous e-commerce websites. This subset contains 24 independent web sources and a total of 29,787 camera product records, each extracted as a flat JSON object composed of attribute–value pairs and a <page title> field. The dataset captures substantial variation across sources in naming conventions, attribute availability, textual representation, and data cleanliness, making it a suitable testbed for evaluating schema alignment and entity-level diagnosis. The attribute space for the camera vertical is large and diverse, comprising 4,660 distinct attribute names arising from differences in vendor formatting, vocabulary, and granularity. Individual sources vary widely in structure: some provide dense and well-formatted specifications, while others contain sparse or noisy attribute sets. On average, each record includes approximately 16.7 non-null attributes, reflecting the heterogeneity inherent in real-world product feeds. This work focuses exclusively on the camera subset because it presents a realistically complex environment for

assessing LLM-assisted data diagnosis, including challenges such as synonym/homonym attributes, inconsistent units, missing or corrupted values, and multi-source redundancy. The camera vertical also includes a manually curated ground truth of 56 mediated attributes, 103 entities, and entity clusters of up to 184 linked records, enabling controlled evaluation of schema-level and instance-level inference.

## 4.2 Application Domain

The application domain is online retail product data, where specifications for consumer electronics such as cameras, are collected from multiple e-commerce platforms. This domain is characterized by heterogeneous attribute formats, inconsistent terminology, and varying data quality, which makes it an ideal setting for evaluating data-diagnosis and integration methods.

## 4.3 Language Model (LM) Configuration

For both schema-matching fine-tuning and prompt-based data imputation, we adopt Qwen2-0.5B [37], a compact yet high-performing language model from the Qwen2 family. Qwen2 models are trained on a large multilingual and multimodal corpus that includes web documents, technical text, code, and diverse structured data sources to create strong generalization across reasoning tasks and domain-shift conditions. The 0.5B-parameter variant balances computational efficiency with sufficient capacity to capture semantic relationships in product-spec data, which makes it suitable for experimentation under limited GPU resources while still offering robust contextual understanding. For schema matching, we perform parameter-efficient fine-tuning using a LoRA configuration. The chosen configuration, r=16, lora_alpha=32, and lora_dropout=0.05, provides a balanced level of expressiveness and regularization, enabling stable learning on heterogeneous attribute pairs. By targeting only the attention projection layers (q_proj, k_proj, v_proj, o_proj), the adaptation focuses on improving how the model attends to and distinguishes attribute names, values, and semantic patterns that are central to schema alignment. Using LoRA in this way keeps training computationally lightweight while allowing the model to specialize effectively for domain-specific matching within the causal language modeling framework.

For embedding-based entity matching, we employ the sentence-transformers/all-MiniLM-L6-v2 model(SBERT) [38], a lightweight sentence embedding model designed to produce dense, semantically meaningful vector representations of short texts. The model is based on a distilled MiniLM architecture with six transformer layers and is trained using a contrastive learning objective to map semantically similar text pairs to nearby points in the embedding space while pushing dissimilar pairs apart. Given a canonical entity signature or an incoming record signature, the model encodes the text into a fixed-dimensional embedding vector, which is subsequently L2-normalized to enable cosine similarity as a stable similarity measure. This representation learning approach captures contextual and semantic similarity beyond exact lexical overlap, making it well suited for entity matching across heterogeneous and noisy attribute descriptions while maintaining low computational overhead.

## 4.4 Evaluation Settings

To evaluate schema matching performance, we assess the Qwen-2.5-0.5B model under two settings: zero-shot inference and fine-tuned adaptation. The evaluation is conducted as an attribute-level classification task, where the model is required to predict the correct target attribute in the canonical schema given a source attribute name. Performance is measured using accuracy, defined as the proportion of correctly matched attribute pairs. To analyze robustness under varying semantic difficulty, attribute pairs are grouped into three categories, as outlined in Table 3: (A) high token overlap, where source and target attributes share strong lexical similarity (e.g., optical zoom → zoom_optical, lens mount → lens_mount_type); (B) synonym or paraphrase relations, where surface forms differ but semantics align (e.g., megapixels → image_resolution, display → screen_size); and (C) noisy or ambiguous cases, where attributes exhibit weak or indirect correspondence (e.g., manufacturer → brand, frame rate → video_resolution). Accuracy is reported separately for each difficulty category, enabling a fine-grained comparison between zero-shot and fine-tuned performance and highlighting the impact of fine-tuning across increasing levels of semantic complexity.

**Table 3: Categories of Difficulty Levels with Examples**

| **Difficulty** | **attribute** | **Target_attribute** |
|---|---|---|
| High Token Overlap (A) | optical zoom | zoom_optical |
| | lens mount | Lens_mount_type |
| Synonym/Paraphrase(B) | megapixels | image_resolution |
| | display | screen_size |
| Noisy/ambiguous (C) | manufacturer | brand |
| | frame rate | Video_resolution |

For entity matching evaluation, we perform a top-1 entity linking experiment against a canonical entity repository using three similarity-based approaches: SBERT embeddings with cosine similarity, TF-IDF representations with cosine similarity, and Jaccard token overlap. A ground-truth mapping between individual specification records and their corresponding canonical entity identifiers is used as reference. For each test record, a single normalized textual signature is constructed and used consistently across all methods to ensure a fair comparison. In the TF-IDF and Jaccard baselines, similarity scores are computed between the query signature and all canonical entity signatures, and the entity with the highest similarity is selected. The SBERT-based method follows the same top-1 retrieval paradigm using embedding similarity and a predefined cosine threshold. Evaluation is conducted using exact-match accuracy, defined as the proportion of records for which the predicted entity identifier matches the ground-truth entity. Precision, recall, and F1-score are additionally reported, where a prediction is considered positive only if it correctly links to the true entity. This evaluation protocol ensures a direct and consistent comparison of embedding-based and lexical similarity methods for entity matching under identical input and decision constraints.

For data imputation evaluation, we assess the effectiveness of three imputation strategies on records with known ground-truth attribute values: (i) entity-aware LLM-based imputation, (ii) entity-level statistical imputation using the mode of observed values, and (iii) a similarity-based kNN baseline without explicit entity constraints. For each target attribute

(e.g., megapixel or screen size), a held-out evaluation set is constructed consisting of JSON records with the corresponding attribute removed, while the true value is retained separately as ground truth. Each record is first resolved to a canonical entity using the entity matching module. In the entity-aware LLM setting, missing values are inferred using prompt-based reasoning over records belonging to the same resolved entity. In the entity-mode baseline, the missing attribute is imputed by selecting the most frequent observed value among entity-consistent records. In the kNN baseline, the attribute is inferred by retrieving the most similar records from the full dataset based on TF-IDF cosine similarity over textual product descriptions and taking the mode of their attribute values. To account for heterogeneous attribute naming, multiple attribute aliases are resolved automatically when aggregating evidence. Imputation performance is evaluated using accuracy, defined as the proportion of correctly imputed values under an exact-match or tolerance-based criterion, together with coverage, defined as the proportion of records for which a method produces a valid imputation. This evaluation protocol enables a direct comparison between LLM-based reasoning and non-LLM baselines under identical entity resolution and ground-truth conditions.

# 5 Results

This section presents and discusses the empirical findings of the proposed framework across schema matching, entity resolution, and data imputation tasks. The results are analyzed to assess the effectiveness of prompt-based LLM reasoning and embedding-driven similarity methods under varying levels of semantic complexity and data sparsity. In particular, we examine how model performance changes across attribute difficulty levels in schema matching, compare embedding-based and lexical baselines for entity resolution, and evaluate the impact of entity-aware prompt-based reasoning for imputing missing attributes.

Fist we have fine tune the Qwen2-0.5B model for schema matching while 4720 data were selected from processed 128k attribute details, due to resource constraint. The model was further validation on validation dataset that contains 500 records. For this evaluation , the model performance at zero shot and fine tune on this validation dataset and the results is reported in Table 4. From this result, we can observe that , Fine-tuning substantially improves schema matching performance, increasing accuracy from 0.16 in the zero-shot setting to 0.87, indicating that task-specific adaptation is critical for reliable attribute alignment.

**Table 4: Model performance over Zero Shot vs Fine Tuned on Validation Dataset**

| | **Zero-shot** | **Fine-Tuned** |
|---|---|---|
| **Accuracy** | 0.16 | 0.87 |

The results illustrated in Figure 3, show a systematic effect of fine-tuning on schema matching performance across all difficulty levels. In the zero-shot setting, the model achieves limited accuracy only in the high token overlap category (≈0.20) and fails entirely on synonym/paraphrase and noisy/ambiguous attributes. This behavior indicates that, without task-specific supervision, the model relies primarily on surface-level lexical overlap and struggles to infer semantic equivalence when attribute names differ in wording or are indirectly related. In contrast, fine-tuning leads to a dramatic improvement across all categories, with high token overlap and synonym/paraphrase attributes reaching near-perfect accuracy (≈0.89

and ≈0.94, respectively). This suggests that fine-tuning enables the model to learn stable mappings between semantically equivalent attributes, even when their surface forms differ substantially, by internalizing domain-specific terminology and alignment patterns. Performance on noisy or ambiguous attributes also improves significantly (≈0.55) but remains lower than in the other categories. This gap is likely due to the inherently underspecified nature of such attributes, where mappings depend on contextual interpretation or implicit assumptions (e.g., frame rate versus video resolution), rather than direct semantic equivalence. Overall, these results demonstrate that fine-tuning is essential for robust schema matching, particularly for resolving synonymy and domain-specific paraphrases, while also highlighting that noisy and ambiguous attribute mappings remain a challenging case even with supervised adaptation.

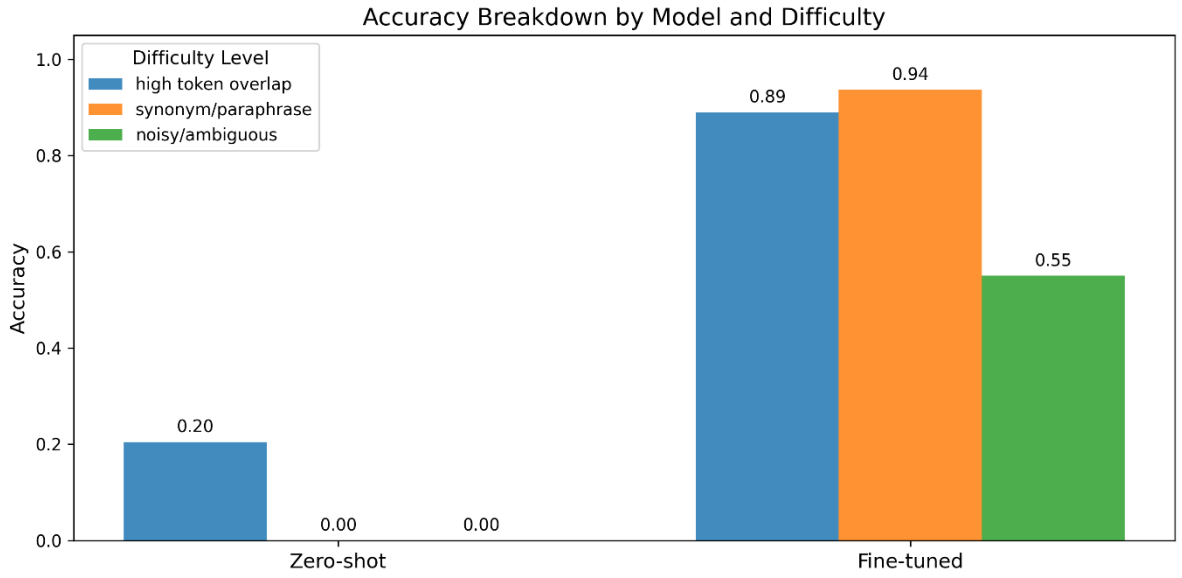



**Figure 3: Schema Matching Accuracy by Attribute Difficulty Level for Zero-Shot and Fine-Tuned Qwen-2.5-0.5B**

Figure 4 presents a comparative evaluation of entity matching performance using three similarity-based methods: SBERT embeddings with cosine similarity, TF-IDF cosine similarity, and Jaccard token overlap. Performance is reported in terms of both accuracy and F1-score to capture exact matching performance as well as robustness to partial or noisy matches. The SBERT-based approach substantially outperforms the lexical baselines, achieving the highest accuracy (0.94) and F1-score (0.97). This result indicates that contextual embeddings are highly effective at capturing semantic similarity between product specifications and canonical entity representations, even when surface-level token overlap is limited. In contrast, the TF-IDF cosine baseline achieves moderate performance (accuracy = 0.50, F1 = 0.67), reflecting its reliance on term frequency and n-gram overlap, which limits its ability to resolve paraphrases or structurally varied descriptions. The Jaccard similarity baseline performs slightly worse (accuracy = 0.46, F1 = 0.63), as it considers only set-level token overlap and ignores both token importance and contextual meaning. Overall, the results demonstrate that embedding-based semantic representations provide a significant advantage for entity matching in heterogeneous and noisy product data, validating the choice of SBERT cosine similarity as the core entity resolution mechanism in the proposed framework.

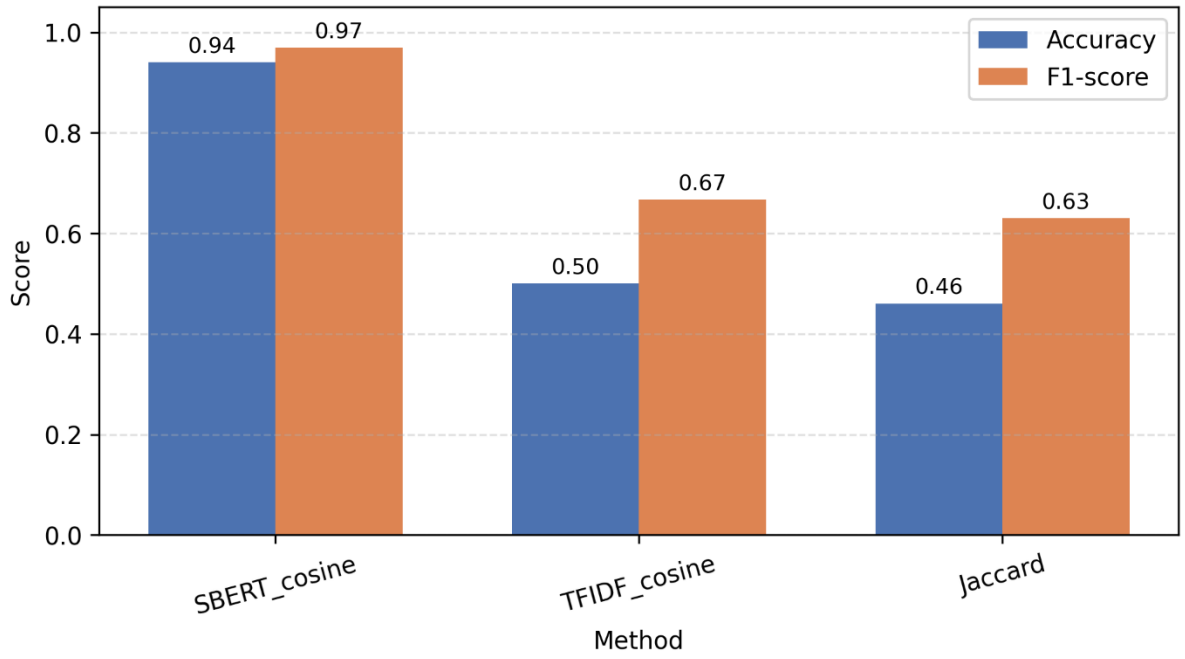



**Figure 4: Entity Matching Performance Comparison across Similarity-based Methods**

Evaluating data imputation in real-world product specifications is challenging due to the limited availability of records with reliable ground-truth values. To construct a controlled evaluation setting, we randomly sampled 12 JSON records from the collected dataset for each target attribute. For the megapixel attribute, pixel-related information was removed from each selected JSON file and retained separately as ground truth. An analogous procedure was applied for the screen size attribute, where screen size information was removed and held out from another randomly selected set of 12 JSON records. Imputation performance was evaluated by comparing the imputed values against the held-out ground truth using accuracy as the primary metric.

**Table 5: Accuracy Comparison of Data Imputation Methods on Held-out Attributes**

| | | llm | mode | knn |
|---|---|---|---|---|
| MegaPixel | Accuracy | 0.17 | 0.08 | 0.08 |
| Screen_Size | Accuracy | 0.25 | 0.25 | 0.27 |

These results indicate that data imputation remains a challenging task under sparse supervision and limited sample size. For the megapixel attribute, the LLM-based imputation approach achieves higher accuracy (0.17) compared to both the entity-mode and kNN baselines (0.08), suggesting that prompt-based reasoning provides some benefit when entity-level evidence is weak or inconsistent. However, overall performance remains low, reflecting the high variability and noise present in megapixel representations across sources. For the screen size attribute, all three methods perform comparably, with accuracy ranging between 0.25 and 0.27. This suggests that screen size information is more consistently represented across records, reducing the relative advantage of LLM-based reasoning over simpler statistical or similarity-

based baselines. These findings highlight both the potential and limitations of entity-aware LLM-based imputation, particularly in low-resource settings with limited ground-truth availability.

# 6 Conclusion

This proposed research presents an integrated framework for schema matching, entity resolution, and data imputation in heterogeneous product specification data, leveraging both embedding-based similarity and prompt-based large language model reasoning. Experimental results demonstrate that fine-tuned language models substantially improve schema matching accuracy across varying levels of attribute difficulty, while embedding-based entity resolution using SBERT and cosine similarity significantly outperforms traditional lexical baselines. These findings confirm the effectiveness of combining semantic representations with task-specific adaptation for resolving structural and semantic heterogeneity in real-world datasets. Despite these strengths, few limitations were observed in this research. Most notably, data imputation remains a challenging component of the framework. Due to the difficulty of constructing reliable evaluation datasets, imputation experiments were conducted on a small number of held-out samples, resulting in relatively low accuracy across all methods. While the LLM-based imputation approach shows modest improvements over statistical and similarity-based baselines in certain cases, overall performance remains limited, highlighting the sensitivity of imputation to data sparsity, inconsistent attribute representations, and limited entity-level evidence. These results indicate that prompt-based reasoning alone is insufficient when supporting data is scarce or noisy. Future work will focus on addressing these limitations through the inclusion of substantially larger and more diverse datasets. Expanding the quantity and coverage of entity-consistent records is expected to significantly improve imputation reliability. Additional future research should include exploring hybrid imputation strategies that combine prompt-based reasoning with numerical normalization, uncertainty modelling, and multi-attribute joint inference. In summary, this study provides a foundational framework and highlights key challenges and provides a clear direction for advancing reliable data integration and completion in complex, real-world settings.

# 7 References


[1] Jason Brownlee. 2020. Data preparation for machine learning: data cleaning, feature selection, and data transforms in Python. Machine Learning Mastery.

[2] Mazhar Hameed and Felix Naumann. 2020b. Data preparation: A survey of commercial tools. ACM SIGMODRecord, 49(3):18–29.

[3] Sheetrit, E., Brief, M., Mishaeli, M., & Elisha, O. (2024). Rematch: Retrieval enhanced schema matching with llms. *arXiv preprint arXiv:2403.01567*.

[4] Roee Shraga, Ofra Amir, and Avigdor Gal. 2021. Learning to Characterize Match ing Experts. In 2021 IEEE 37th International Conference on Data Engineering (ICDE). 1236–1247. https://doi.org/10.1109/ICDE51399.2021.00111

[5] Chen, M., Sun, Y., Li, T., Wang, J., Wang, K., Lin, X., ... & Zhang, W. (2025). Empowering Tabular Data Preparation with Language Models: Why and How?. *arXiv preprint arXiv:2508.01556*.

[6] Nan Tang, Ju Fan, Fangyi Li, Jianhong Tu, Xiaoyong Du, Guoliang Li, Sam Madden, and Mourad Ouzzani. Rpt: relational pre-trained transformer is al most all you need towards democratizing data preparation. arXiv preprint arXiv:2012.02469, 2020.

[7] Chengliang Chai, Guoliang Li, Ju Fan, and Yuyu Luo. Crowdchart: Crowd sourced data extraction from visualization charts. IEEE Trans. Knowl. Data Eng., 33(11):3537–3549, 2021.

[8] Xuanhe Zhou, Guoliang Li, Chengliang Chai, and Jianhua Feng. A learned query rewrite system using monte carlo tree search. Proc. VLDB Endow., 15(1):46–58, 2021.

[9] Minghua Ma, Zheng Yin, Shenglin Zhang, and et al. Diagnosing root causes of intermittent slow queries in large-scale cloud databases. Proc. VLDB Endow., 13(8):1176–1189, 2020.

[10] Zhou, X., Zhao, X., & Li, G. (2024). Llm-enhanced data management. *arXiv preprint arXiv:2402.02643*.

[11] Avanika Narayan, Ines Chami, Laurel J. Orr, and Christopher R'e. 2022. Can Foundation Models Wrangle Your Data? Proc. VLDB Endow. 16 (2022), 738–746. https://api.semanticscholar.org/CorpusID:248965029

[12] Haochen Zhang, Yuyang Dong, Chuan Xiao, and M. Oyamada. 2023. Large Language Models as Data Preprocessors. ArXiv abs/2308.16361 (2023). https: //api.semanticscholar.org/CorpusID:261397017

[13] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, *33*, 1877-1901.

[14] Liu, Y., Han, T., Ma, S., Zhang, J., Yang, Y., Tian, J., ... & Ge, B. (2023). Summary of chatgpt-related research and perspective towards the future of large language models. *Meta-radiology*, *1*(2), 100017.

[15] Mirchandani, S., Xia, F., Florence, P., Ichter, B., Driess, D., Arenas, M. G., ... & Zeng, A. (2023). Large language models as general pattern machines. *arXiv preprint arXiv:2307.04721*.

[16] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving Language Understanding by Generative Pre-Training. https://api. semanticscholar.org/CorpusID:49313245

[17] Neelakantan, A., Xu, T., Puri, R., Radford, A., Han, J. M., Tworek, J., ... & Weng, L. (2022). Text and code embeddings by contrastive pre-training. *arXiv preprint arXiv:2201.10005*.

[18] Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021). Deep learning--based text classification: a comprehensive review. *ACM computing surveys (CSUR)*, *54*(3), 1-40.

[19] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to Fine-Tune BERT for Text Classification?. In China National Conference on Chinese Compu tational Linguistics. https://api.semanticscholar.org/CorpusID:153312532

[20] Sun, C., Qiu, X., Xu, Y., & Huang, X. (2019, October). How to fine-tune bert for text classification?. In *China national conference on Chinese computational linguistics* (pp. 194-206). Cham: Springer International Publishing.

[21] Parciak, M., Vandevoort, B., Neven, F., Peeters, L. M., & Vansummeren, S. (2024). Schema matching with large language models: an experimental study. *arXiv preprint arXiv:2407.11852*.

[22] Ma, Z., Zhang, B., Zhang, J., Yu, J., Zhang, X., Zhang, X., ... & Tang, J. (2024). Spreadsheetbench: Towards challenging real world spreadsheet manipulation. *Advances in Neural Information Processing Systems*, *37*, 94871-94908.

[23] Feng, L., Li, H., & Zhang, C. J. (2024). Cost-Aware Uncertainty Reduction in Schema Matching with GPT-4: The Prompt-Matcher Framework. *arXiv e-prints*, arXiv-2408.

[24] Liu, Y., Pena, E., Santos, A., Wu, E., & Freire, J. (2024). Magneto: Combining small and large language models for schema matching. *arXiv preprint arXiv:2412.08194*.

[25] Seedat, N., & van der Schaar, M. (2024). Matchmaker: Self-improving large language model programs for schema matching. *arXiv preprint arXiv:2410.24105*.

[26] Wu, S., Wu, Q., Dong, H., Hua, W., & Zhou, X. (2023). Blocker and matcher can mutually benefit: a co-learning framework for low-resource entity resolution. *Proceedings of the VLDB Endowment*, *17*(3), 292-304.

[27] Peeters, R., Steiner, A., & Bizer, C. (2023). Entity matching using large language models. *arXiv preprint arXiv:2310.11244*.

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

[29] Li, H., Li, S., Hao, F., Zhang, C. J., Song, Y., & Chen, L. (2024, May). Booster: Leveraging large language models for enhancing entity resolution. In *Companion Proceedings of the ACM Web Conference 2024* (pp. 1043-1046).

[30] Peeters, R., & Bizer, C. (2021). Dual-objective fine-tuning of BERT for entity matching. *Proceedings of the VLDB Endowment*, *14*, 1913-1921.

[31] Li, Y., Li, J., Suhara, Y., Doan, A., & Tan, W. C. (2020). Deep entity matching with pre-trained language models. arXiv preprint arXiv:2004.00584.

[32] Steiner, A., Peeters, R., & Bizer, C. (2024). Fine-tuning large language models for entity matching. *arXiv preprint arXiv:2409.08185*.

[33] He, X., Ban, Y., Zou, J., Wei, T., Cook, C. B., & He, J. (2024). LLM-Forest for Health Tabular Data Imputation. *arXiv e-prints*, arXiv-2410.

[34] Ding, Z., Tian, J., Wang, Z., Zhao, J., & Li, S. (2024). Data imputation using large language model to accelerate recommendation system. *arXiv preprint arXiv:2407.10078*.

[35] Wang, T., Chen, X., Lin, H., Chen, X., Han, X., Sun, L., ... & Zeng, Z. (2025, January). Match, compare, or select? an investigation of large language models for entity matching. In *Proceedings of the 31st International Conference on Computational Linguistics* (pp. 96-109).

[36] Crescenzi, V., De Angelis, A., Firmani, D., Mazzei, M., Merialdo, P., Piai, F., & Srivastava, D. (2021). Alaska: A flexible benchmark for data integration tasks. *arXiv preprint arXiv:2101.11259*.

[37] https://huggingface.co/Qwen/Qwen2-0.5B

[38] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

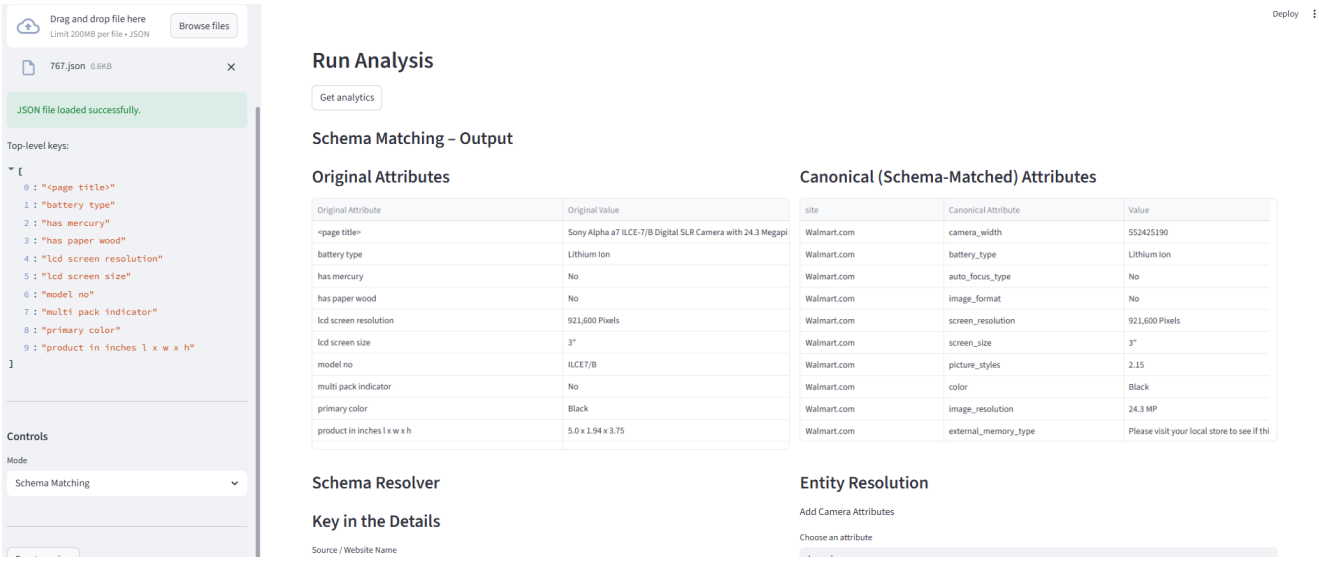
# 8 Appendix

Selected Common Core Attributes of Records

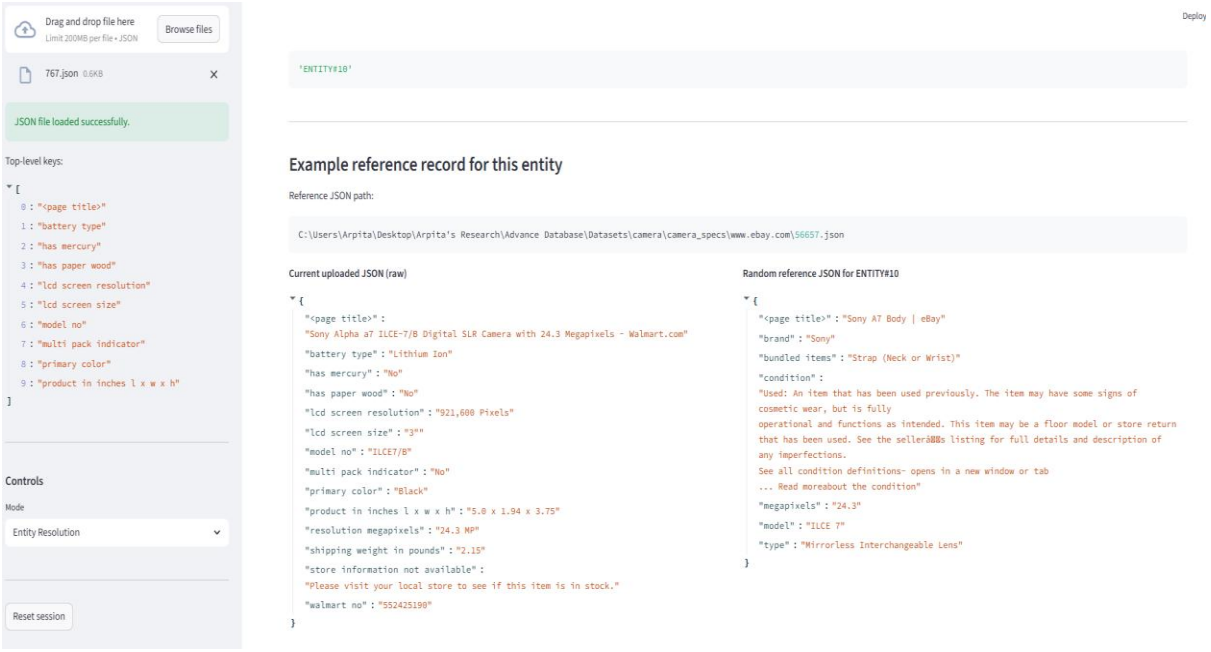
| Main_Attribute_Name | Rrepresentations |
|---|---|
| brand | "brand", "manufacturer", "maker", "vendor", "company" |
| model | "model", "product name", "name", "product", "title", "page title", "model number", "model no", "mfr part", "part number", "item model", "sku", "product id" |
| sensor | "sensor", "sensor type", "image sensor", "sensor size", "imaging sensor", "cmos", "ccd" |
| lens | "lens", "lens type", "focal length", "kit", "optical zoom", "zoom lens", "lens configuration", "lens model", "lens description" |
| megapixel | "megapixel", "mp", "effective megapixels", "resolution", "total pixels" |
| Camera_type | "camera type", "dslr", "slr", "mirrorless", "compact camera","point and shoot", "bridge camera", "system camera", "digital camera" |
| Body_info | "body only", "camera body only", "kit", "camera kit", "with lens", "with 18-55mm lens", "with 28-70mm lens" |

**Model Deployment Dashboard**

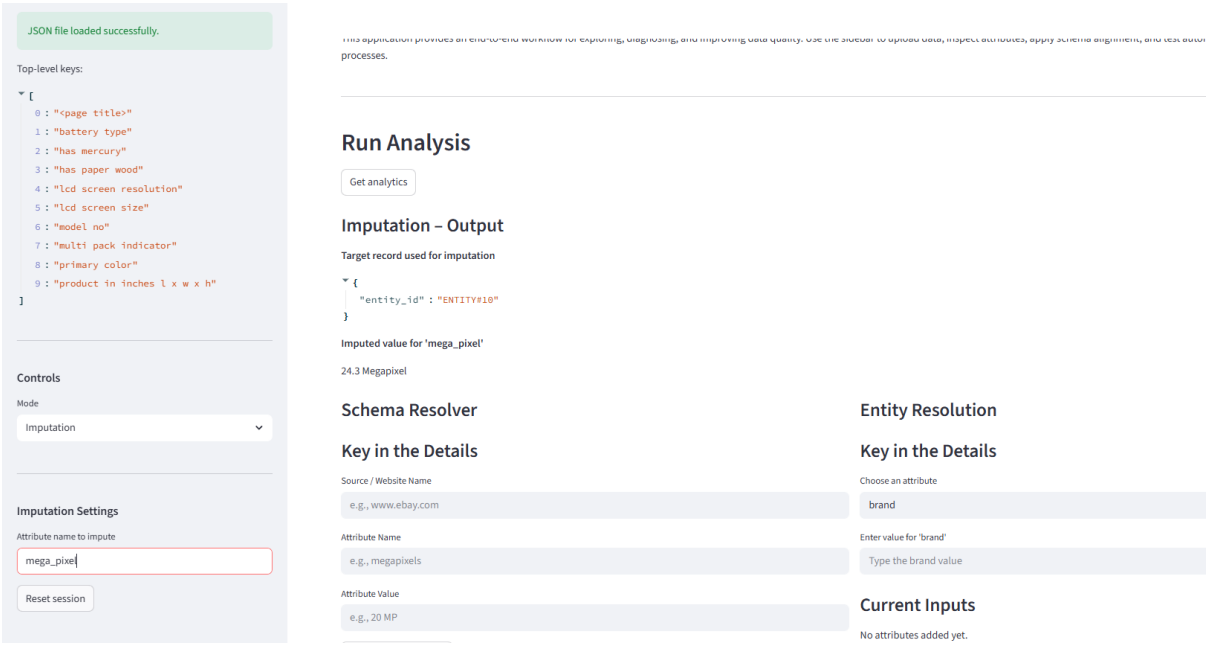
**Schema Matching**



## Entity Matching




## Data Imputation



## GitHub Repo

https://github.com/Arpi33/Advance_Database.git

## Declaration of AI Tool Usage

In a limited number of places, I have used AI tool only to improve clarity and rephrase the writing style. All methodological design, implementation, experimentation, analysis, and conclusions presented in this work were developed entirely by me.