CLEAR: Contextual Learning based Explanations for Anomaly Reasoning

Anonymous Authors[1]

Abstract

Erroneous or fraudulent invoices present significant risks to financial operations in online marketplaces, and anomaly detection offers a better solution to mitigate those risks. Despite advances in machine learning-based anomaly detection, the black-box nature of these models limits their adoption in Finance, where manual review is required. Human investigators often struggle to review numerous flagged invoices due to the absence of clear, contextual explanations, resulting in only 40% of true defects being detected by investigator. We propose CLEAR, a multi stage model-agnostic framework that combines contrastive learning and large language models (LLMs) to generate context-rich, human-readable explanations. CLEAR projects anomalous examples into a latent space to find semantically similar, non-anomalous counterparts and identifying key distinguishing features using localized interpretable models. These features are passed to a context-aware LLM fine-tuned with historical investigator feedback to generate concise summaries, improving investigation efficiency from 40% to 50% and enabling estimated substantial annual savings while providing interpretability through real-case comparisons and contextual semantics.

1. Introduction

Finance teams at large firms can process hundreds of millions of invoices each year, amounting to billions of dollars in payments to vendors. While the vast majority of these transactions are accurate, a small yet critical fraction—are canceled due to vendor errors, internal processing mistakes, or instances of fraud and abuse. This seemingly minor percentage translates into over a billion dollars in potential financial exposure annually, posing significant risks from both economic and compliance standpoints. These risks also evolve over time, making early detection of anomalous invoices essential to prevent any fraud and abuse from system and to safeguard the integrity of financial operations. In response, major enterprises increasingly leverage machine learning (ML) models to proactively flag invoices that may be erroneous or fraudulent. These models often built using deep learning or ensemble methods are highly effective at identifying subtle statistical patterns in large, high-dimensional financial datasets. However, their predictive power often comes at the cost of interpretability. As a result, these models operate as black boxes, producing predictions without offering clear explanations.

In the financial domain, this lack of transparency poses a serious challenge. Regulatory and compliance requirements prohibit fully automated actions based solely on model predictions. Every flagged invoice must be reviewed and justified by human investigators before cancellation. Yet, investigators are often left with little more than opaque feature scores, lacking the context needed to make informed decisions under tight deadlines and audit scrutiny.

This gap between model performance and human interpretability has led to a significant operational bottleneck. From the set of flagged invoices, only 40% of true defects are correctly identified and acted upon by investigators. The remaining 60% despite being flagged are missed due to lack of explanation and actionable insights. Bridging this gap is essential for realizing the full potential of ML-driven invoice risk detection in high-stakes financial environments.

Traditional model-agnostic explainability techniques, such as SHAP (5) and LIME (7), have been widely used to interpret black-box predictions by estimating the marginal contribution of individual features through local perturbation-based approximations. These methods are effective but present critical shortcomings in financial systems like fraud detection (10). Specifically, they operate under the assumption of feature independence and rely on synthetic perturbations in the input space that may violate the underlying data manifold (4). As a result, they often generate explanations that are mathematically plausible but semantically invalid, perturbing a vendor type or invoice amount in isolation, without regard for inter-feature dependencies or business rules (e.g., approval status conditional on amount or supplier type). Moreover, their outputs are typically unstructured importance scores across raw feature names, lacking semantic enrichment or narrative grounding. For example, stating

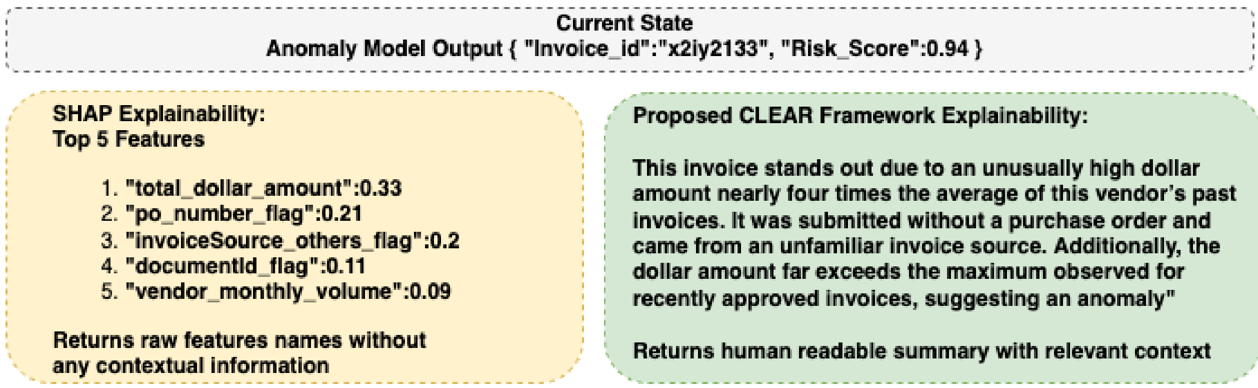

Figure 1. CLEAR vs Traditional Explainability Approaches

that "feature_132 increased risk by 0.18" offers little actionable insight to a financial investigator tasked with assessing fraud.

This paper makes the following core contributions:

- **Model-Agnostic Local Explainability Framework:** A novel plug-and-play interpretability pipeline that can be layered on top of any anomaly detection model regardless of its internal complexity—to produce clear, localized explanations without modifying the underlying predictive architecture.
- **Real-Data Counterfactuals with Auditable Chains:** A novel approach to counterfactual explanation that retrieves semantically similar, historically observed normal cases via learned embeddings, preserving the data manifold and yielding deterministic, fully traceable explanation chains rooted in actual examples.
- **LLM-Driven Context-Aware Justifications:** An explanation module that ingests features from a metadata-enhanced surrogate model and leverages a large language model to generate concise, human-readable explanations aligned with domain semantics and investigator intuition.

By integrating structured similarity search, local explainable modeling, and contextualized language generation, we bridge the gap between machine learning detection and human decision making, a necessary step toward trustworthy AI in finance.

2. Related Works

Interpretable Models and Post-hoc Explainability: Traditional models like logistic regression and decision trees are transparent but limited in capturing complex patterns, SHAP(5) explains autoencoder reconstruction errors (1), while LIME (7) uses local surrogates. Anchors (8) provide high-precision if-then rules. These methods rely on synthetic perturbations and often ignore contextual or temporal dependencies. Comparison with CLEAR is in Table 1.

Embedding-based Retrieval for Explanation: Instance-based explanations using embeddings show promise. Dang et al. (3) used graph embeddings to retrieve similar fraud cases but may miss transactional nuances.

Contrastive Learning (CL) for Anomaly Interpretation: CL improves anomaly detection by separating normal and anomalous cases. Zhang et al. (9) showed its utility in fraud detection, though it requires careful sampling and faces class imbalance issues.

LLM-based Explanation Generation: LLMs generate natural language explanations. Park et al. (6) translated anomaly scores into narratives, but face hallucination, weak data grounding, and audit challenges.

Summary and Positioning: CLEAR combines contrastive embeddings, local interpretable models, and domain-adapted LLMs for grounded, auditable, and regulatory-aligned explanations.

Table 1. Comparison of CLEAR with SHAP and LIME

Aspect	**SHAP/LIME**	**CLEAR**
Perturbations	Synthetic	None (real data)
Global vs Local	Local	Local
Grounded in Training Data	No	Yes
Hallucination Risk	High	Minimal (prompt control)
Auditability	Weak	Strong

3. Methodology

CLEAR is a model-agnostic, post-hoc interpretability framework designed to layer on top of any anomaly detection model. It generates human-interpretable, audit-ready, and context-rich explanations for anomalous instances. The framework comprises four sequential stages (Figure 2) as follows:

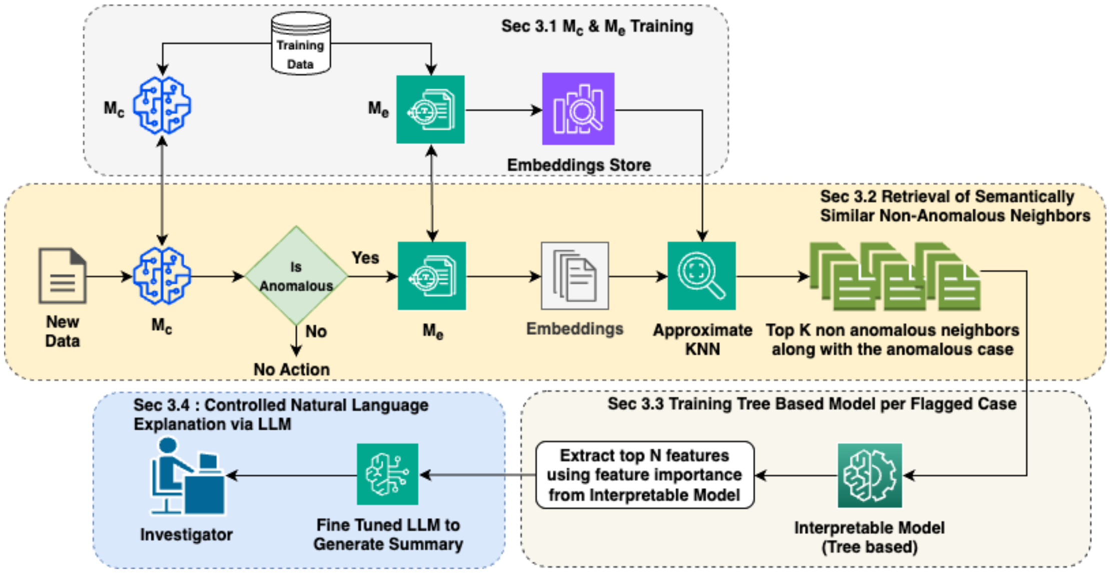

Figure 2. CLEAR framework

3.1. Embedding and Anomaly Model Training

We first train an anomaly detector M_c on historical data X_h to assign anomaly scores to new inputs. Since traditional distance-based retrieval methods on high-dimensional tabular data suffer from poor discrimination, we train a separate embedding model M_e (e.g., using contrastive learning) to capture semantic and contextual patterns. After training, M_e encodes all non-anomalous instances into a dense embedding space, which serves as the retrieval corpus.

3.2. Neighbor Retrieval via Approximate KNN

For each data point flagged as anomalous by M_c, we generate its embedding using M_e and retrieve its top-K most similar normal cases via approximate K-nearest neighbor search. These neighbors form a semantically and behaviorally grounded peer group for contrastive reasoning.

3.3. Local Surrogate Modeling and Feature Attribution

A local surrogate model (e.g.Random forest) is trained using the K retrieved neighbors and the flagged anomaly to approximate M_c's local decision boundary. We extract the top-N discriminative features that distinguish the anomaly from its peer group for further step.

3.4. Controlled Natural Language Explanation via LLM

Finally, a domain-adapted LLM uses the extracted N features and contextual neighbors to generate structured, natural language justifications. Fine-tuned with domain knowledge and contextual understanding between features, the LLM ensures explanations are grounded, audit-friendly, and semantically aligned with model behavior.

4. Dataset and Experimental Setup

4.1. Dataset

We use a proprietary dataset of six million structured business invoices, each with raw attributes like amount, date, source, approval status, and payment terms. From these, 210 derived features are created for the anomaly detector, capturing current invoice properties (e.g., PO presence, source legitimacy) and historical vendor behavior (e.g., transaction volume, payment trends etc.) for rich contextual modeling.

4.2. Experimental Setup

CLEAR is tested on this tabular dataset to produce audit-ready explanations for flagged invoices.

Anomaly Detector (M_c): A deep learning classifier trained on the 210 features to score invoices by fraud likelihood. It captures both cross-sectional and temporal patterns in vendor-invoice interactions.

Embedding Model (M_e): To learn contextual representations, we employ SCARF (2), a contrastive learning method with random feature corruption (See Figure 3 in Appendix 7.2 for embedding model architecture). The architecture consists of: (1) an MLP M for encoding categorical features, (2) an encoder E, and (3) a projection head H. Each input x_i and its corrupted version x_i^+ are passed through these layers to yield embeddings z_i and z_i^+:

$$z_i = g_\phi(x_i) = H(E(x_i + M(x_i))), \quad z_i^+ = g_\phi(x_i^+)$$

The model minimizes the InfoNCE loss:

$$L_{\mathrm{ctr}}(\phi) = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(\mathrm{sim}(z_i, z_i^+)/\tau)}{\sum_{j=1}^{N}\exp(\mathrm{sim}(z_i, z_j)/\tau)}$$

where $\mathrm{sim}(\cdot,\cdot)$ is cosine similarity and τ is the temperature.

Neighbor Retrieval: In production, if M_c flags an invoice, it is encoded via M_e, and the top-K ($K = 200$) most similar non-anomalous embeddings are retrieved using approximate KNN. These neighbors define a context-aware peer group for interpretation.

Local Surrogate Modeling: A Random Forest classifier is trained on the flagged invoice and its K neighbors to identify key discriminative features. The top $N = 5$ features based on importance scores form the explanation vector. These K, N values were empirically tuned for optimal fidelity.

LLM-based Explanation: We fine-tuned Gemma-7b on 50K labeled invoices with investigator notes and feature metadata. At inference, the top-N features and metadata are passed to the LLM, which generates structured, fluent explanations grounded in domain-specific semantics, closing the loop from detection to interpretation.

5. Results

We have reported CLEAR framework's performance for three different settings : Responsiveness to counterfactuals, alignment score, and impact on downstream decision-making using A/B testing.

- **Responsiveness to counterfactuals** is defined as the consistency and sensitivity of the explanation method to minimal changes in the input features. A method is deemed responsive if small, plausible perturbations to the input lead to coherent changes in the top-N features identified as most important. We tested this across 1000 randomly selected invoices where we introduced targeted counterfactual edits (e.g., modifying the invoice amount or vendor history flags). CLEAR exhibited a 22.4% & 10.5% relative improvement in responsiveness compared to the SHAP+LLM & LIME+LLM respectively. This suggests that CLEAR captures better nuanced shifts in feature interactions.

- **Alignment Score**: To measure how useful the explanations were to actual users, we introduced the Alignment Score metric. The alignment score captures how well the top-N features identified by each method match the investigator's own reasoning. Based on investigator notes, we extracted the most critical features for each case (blind to model output) and scored overlap via the Jaccard index between annotated and suggested features. CLEAR achieved a 22.4% and 15.5% relative improvement in alignment score over the SHAP+LLM and LIME+LLM respectively. The Jaccard Index is defined as:

  $$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

  where A is set of top-N features identified by the explanation method & B is set of critical features extracted from the investigator notes.

- **A/B testing**: We conducted an **A/B test** across control and three treatment groups of investigators to assess the real-world impact of CLEAR. The test results demonstrated a statistically significant (p = 0.02) 25% relative improvement in investigator efficiency. The A/B test provides clear evidence that CLEAR enhances both the quality of explanations (more faithful and readable) and the practical results of investigations.

Investigator Efficiency: Investigator Efficiency (IE), analogous to recall in machine learning, measures the proportion of cancelled invoices correctly identified by investigators out of all cancelled invoices in the model-flagged sample. It reflects how effectively investigators detect true cancellations among flagged invoices.Investigator Efficiency is formally defined as:

$$\text{Investigator Efficiency} = \frac{C}{N} \quad (2)$$

where: C is the number of cancelled invoices correctly identified and N is total cancelled invoices in the flagged sample.

Table 2. Comparison of Explanation Methods

Method	**Resp. (%)**	**Align. Score**	**Eff. (%)**
Control/Baseline	-	-	40.2
CLEAR	**93.0**	**0.71**	**50.1**
SHAP + LLM	76.0	0.58	44.3
LIME + LLM	68.0	0.49	42.9

6. Conclusion and Future Works

In this paper, we proposed CLEAR, a comparison-based explainability framework that distinguishes anomalous from normal cases by comparing them directly to historical instances instead of synthetic perturbations. It addresses key limitations of SHAP (5) and LIME (7), which often produce explanations lacking alignment with human reasoning. CLEAR offers interpretable explanations that resonates with how investigators reason through known examples. Future work will focus on scaling CLEAR to heterogeneous datasets. By bridging model performance and interpretability, CLEAR enables more actionable and trustworthy AI in enterprise applications.

References

[1] Liat Antwarg, Ronnie Mindlin Miller, Bracha Shapira, and Lior Rokach. Explaining anomalies detected by autoencoders using shapley additive explanations. *Expert systems with applications*, 186:115736, 2021.

[2] Dara Bahri, Heinrich Jiang, Yi Tay, and Donald Metzler. Scarf: Self-supervised contrastive learning using random feature corruption. *arXiv preprint arXiv:2106.15147*, 2021.

[3] Quang-Vinh Dang and Ngoc-Son-An Nguyen. Evaluating the contribution of relationship information in detecting fraud using graph neural networks. In *Inventive communication and computational technologies: Proceedings of ICICCT 2022*, pages 865–875. Springer, 2022.

[4] Sachin Goyal, Aditi Raghunathan, Moksh Jain, Harsha Vardhan Simhadri, and Prateek Jain. Drocc: Deep robust one-class classification, 2020.

[5] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[6] Taejin Park. Enhancing anomaly detection in financial markets with an llm-based multi-agent framework. *arXiv preprint arXiv:2403.19735*, 2024.

[7] Marco Tulio Ribeiro, Sameer Singh, and Carlos 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.

[8] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Nothing else matters: Model-agnostic explanations by identifying prediction invariance, 2016.

[9] Jiaqiang Zhang, Senzhang Wang, and Songcan Chen. Reconstruction enhanced multi-view contrastive learning for anomaly detection on attributed networks. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, IJCAI-2022, page 2376–2382. International Joint Conferences on Artificial Intelligence Organization, July 2022.

[10] Ying Zhou, Haoran Li, Zhi Xiao, and Jing Qiu. A user-centered explainable artificial intelligence approach for financial fraud detection. *Finance Research Letters*, 58:104309, 2023.

7. Appendix

7.1. CLEAR-Generated Natural Language Explanations with Structured Feature Comparison

Example 1

LLM Prompt:

Human: <COT_prompt>

<model_output>


```
{
  total_dollar_amount: {
    feature_importance: 0.42,
    flagged_invoice: 187000,
    non_flagged_invoices_avg: 49245
  },
  po_number_flag: {
    feature_importance: 0.21,
    flagged_invoice: 0,
    non_flagged_invoices_mode: 1
  },
  invoiceSource_others_flag: {
    feature_importance: 0.14,
    flagged_invoice: 1,
    non_flagged_invoices_mode: 0
  },
  wfapproval_yes_flag_dollar_amount: {
    feature_importance: 0.12,
    flagged_invoice: 52000,
    non_flagged_invoices_avg: 18010
  },
  documentId_flag: {
    feature_importance: 0.11,
    flagged_invoice: 1,
    non_flagged_invoices_mode: 1
  }
}
```


<\model_output>
Assistant:

LLM Output:

"This invoice stands out due to an unusually high dollar amount—nearly four times the average of this vendor's past invoices. It was submitted without a purchase order and came from an unfamiliar invoice source. Additionally, the dollar amount far exceeds the maximum observed for recently approved invoices, suggesting an anomaly."

Example 2

LLM Prompt:

Human: <COT_prompt>

<model_output>

```
{
  invoiceDate_to_invoiceCreationDate: {
    feature_importance: 0.38,
    flagged_invoice: 25,
    non_flagged_invoices_avg: 2
  },
  workflow_approval: {
    feature_importance: 0.25,
    flagged_invoice: NOT_REQUIRED,
    non_flagged_invoices_mode: REQUIRED
  },
  vendor_monthly_volume: {
    feature_importance: 0.18,
    flagged_invoice: very_low,
    non_flagged_invoices_mode: moderate
  },
  documentId_flag: {
    feature_importance: 0.10,
    flagged_invoice: 0,
    non_flagged_invoices_mode: 1
  },
  po_number_flag: {
    feature_importance: 0.09,
    flagged_invoice: 1,
    non_flagged_invoices_mode: 1
  }
}
```

<\model_output>
Assistant:

LLM Output:

"This invoice shows an abnormally long delay between the invoice date and its creation, which is inconsistent with usual processing timelines. It bypassed workflow approval and came from a vendor with minimal historical volume, raising concerns of a one-off or suspicious transaction. The absence of a document ID further weakens its audit trail."

Example 3

LLM Prompt:

Human: <COT_prompt>

<model_output>

```
{
  payment_method_WIRE_flag: {
    feature_importance: 0.33,
    flagged_invoice: 1,
    non_flagged_invoices_mode: 0
  },
  supplier_type_code_TAX_AUTH_flag: {
    feature_importance: 0.27,
    flagged_invoice: 1,
    non_flagged_invoices_mode: 0
  },
  paymentFlag_N_flag_dollar_amount: {
    feature_importance: 0.21,
    flagged_invoice: 99000,
    non_flagged_invoices_avg: 14593
  },
  ofaCountryCode_others_flag: {
    feature_importance: 0.11,
    flagged_invoice: 1,
    non_flagged_invoices_mode: 0
  },
  documentId_flag: {
    feature_importance: 0.08,
    flagged_invoice: 1,
    non_flagged_invoices_mode: 1
  }
}
```

<\model_output>
Assistant:

LLM Output:

"This invoice was paid via wire transfer—a method rarely used for this vendor type. It's associated with a tax authority but shows a high unpaid amount, which is atypical. The country code is also marked as 'others,' adding to the irregularity of this transaction."

Example 4

LLM Prompt:

Human: <COT_prompt>

<model_output>

```
{
  terms_date_to_invoiceCreationDate: {
    feature_importance: 0.29,
    flagged_invoice: -20,
    non_flagged_invoices_avg: 5
```

```
    },
    holdCounts_flag_dollar_amount: {
      feature_importance: 0.23,
      flagged_invoice: 300,
      non_flagged_invoices_avg: 7580
    },
    business_group_Corp_flag: {
      feature_importance: 0.18,
      flagged_invoice: 1,
      non_flagged_invoices_mode: 0
    },
    invoiceDescription_flag: {
      feature_importance: 0.17,
      flagged_invoice: 0,
      non_flagged_invoices_mode: 1
    },
    po_number_flag: {
      feature_importance: 0.13,
      flagged_invoice: 1,
      non_flagged_invoices_mode: 1
    }
  }
```

<\model_output>

Assistant:

LLM Output:

> "This invoice reflects a negative gap between the payment terms and invoice creation dates, an uncommon pattern suggesting backdating or early terms application. Despite a very low dollar value, it was placed on hold, and lacks a description, both of which are not standard for corporate group invoices."

7.2. Embedding Model Architecture

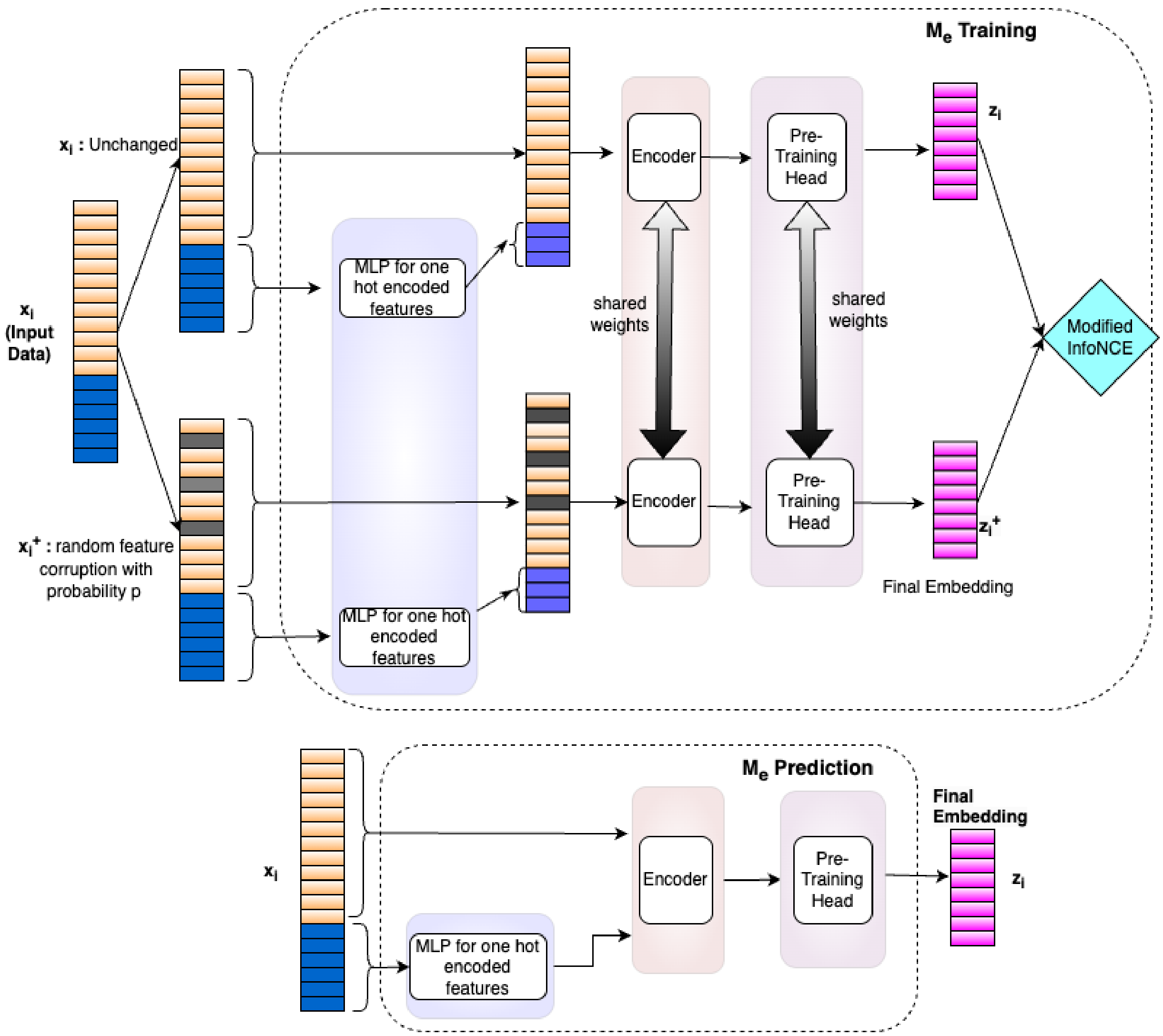

Figure 3. Embedding Model (M_e) Training and Inference Architecture