LEKA: LLM-Enhanced Knowledge Augmentation

Xinhao Zhang[1], **Jinghan Zhang**[1], **Fengran Mo**[2], **Dongjie Wang**[3], **Yanjie Fu**[4] and **Kunpeng Liu**[1*]
[1]Portland State University, USA
[2]University of Montreal, Canada
[3]University of Kansas, USA
[4]Arizona State University, USA
{xinhaoz, jinghanz, kunpeng}@pdx.edu, fengran.mo@umontreal.ca, wangdongjie@ku.edu, yanjie.fu@asu.edu

Abstract

Humans excel in analogical learning and knowledge transfer and, more importantly, possess a unique understanding of identifying appropriate sources of knowledge. From a model's perspective, this presents a unique challenge. If models could autonomously retrieve knowledge relevant for transfer or decision-making to solve problems, they would transition from passively acquiring to actively accessing and learning from knowledge. However, filling models with knowledge is relatively straightforward—it simply requires more training and accessible knowledge bases. The more complex task is teaching models about which knowledge can be analogized and transferred. Therefore, we design a knowledge augmentation method, LEKA, for knowledge transfer that actively searches for suitable knowledge sources that can enrich the target domain's knowledge. This LEKA method extracts key information from the target domain's textual information, retrieves pertinent data from external data libraries, and harmonizes retrieved data with the target domain data in feature space and marginal probability measures. We validate the effectiveness of our approach through extensive experiments across various domains and demonstrate significant improvements over traditional methods in automating data alignment and optimizing transfer learning outcomes.

1 Introduction

Humans are good at identifying relevant sources of knowledge. This is an ability that is rooted in our capability for analogical reasoning and knowledge management. In contrast, artificial intelligence models do not inherently possess this intuition: they require explicit instructions and systematic training to identify and utilize relevant information. This gap presents a challenge in enriching domain knowledge and enhancing data augmentation.

Knowledge augmentation is vital for improving model performance, especially in domains with limited information or complex data structures [Tang *et al.*, 2020]. Knowledge transfer is a crucial method within knowledge augmentation for improving learning performance by transferring knowledge from external information sources [Khodaee *et al.*, 2024]. In data-limited scenarios, effective knowledge sourcing can bridge domain information gaps and enhance model robustness using relevant external information. In these scenarios, knowledge augmentation, especially knowledge transfer, can reduce reliance on extensive target domain data by strategically selecting sources matching target needs.

Despite its potential, traditional knowledge augmentation methods often involve manual intervention [Ringwald and Stiefelhagen, 2021; Nam *et al.*, 2024; Zhang *et al.*, 2024d], where human experts select source domains based on their subjective interpretation of the target domain's requirements. Using tabular learning as an example highlights several challenges: (1) structural and format differences across domains hinder data alignment and integration; (2) discrepancies in tasks and content between the selected source and target domains can diminish the success of knowledge augmentation; and (3) extensive data preprocessing is often required to properly match the chosen source domain dataset. Additionally, relying on human expertise for source domain selection can lead to biases and inefficiencies, as these choices are typically made based on prior knowledge rather than a rigorous, data-driven analysis.

A natural idea is to construct an automated search for datasets with relevant knowledge in a database that has a similar structure to the target data, such as utilizing Retrieval-Augmented Generation (RAG) [Gao *et al.*, 2023; Lewis *et al.*, 2020]. RAG combines retrieval systems with generative language models, and it enhances the model's capabilities by providing access to an external library [Guo *et al.*, 2017; Zhang *et al.*, 2025b; Shi *et al.*, 2018]. For dataset retrieval, some existing works [Fleischer *et al.*, 2024; Siriwardhana *et al.*, 2023] use the retrieved documents as contextual information for generation and significantly enrich the model's knowledge base.

However, this direct method is often impractical due to the high costs of embedding entire domains or

*Corresponding author.

datasets [Seemakhupt *et al.*, 2024; Jin *et al.*, 2024]. Additionally, retrieved knowledge may not align effectively with the intended application [Edge *et al.*, 2024; Li and Ramakrishnan, 2025]. Such misalignments typically result from variations in data distribution, feature spaces, or contextual differences between source and target data. Furthermore, retrieved data quality, like noise or incomplete features, can also hinder its effectiveness for augmenting data via knowledge augmentation. Thus, in such situations, identifying the optimal knowledge source providing the most relevant, high-quality data tailored to specific target data needs via LLM capabilities remains a substantial challenge.

Our Targets. We aim to address three main challenges in retrieving source domain data: (1) how to effectively extract key information from the target domain with limited computational cost; (2) how to design an efficient, automatic source domain retrieval for knowledge augmentation; and (3) how to automatically harmonize the retrieved source data with the target domain to optimize transfer efficiency and improve learning outcomes.

Our Approach. To address the challenges above, we design **L**LM-**E**nhanced **K**nowledge **A**ugmentation (**LEKA**), a novel and automated data retrieval and augmentation method by knowledge augmentation. Specifically, (1) we utilize an LLM to extract the key textual information of the target domain; (2) we deploy dataset-RAG in an external database to efficiently extract knowledge relevant to the target domain dataset. The RAG libraries can contain large amounts of continuously updated datasets, ensuring that the data retrieved from these libraries is more precise and timely; (3) then the LLM automatically harmonizes the retrieved source data with the target domain in feature space and marginal probability measures to enhance downstream task learning performance. The data harmonization reduces the structural and semantic differences between the source and target data. This data harmonization is crucial for reducing domain shift, which optimizes transfer efficiency and improves the overall learning outcomes of the model.

As shown in Figure 1, consider a target domain focused on predicting a rare cancer, so direct learning is challenging due to insufficient sample information. To augment the data with external information would typically require human experts with knowledge of both cancer physiological indicators and machine learning to retrieve datasets. In contrast, our method can automate the retrieval of a common cancer dataset as the knowledge source with extensive feature information. This retrieved dataset possesses a more complete and transparent feature space with a structure similar to the target domain. The LLM then adjusts the retrieved data based on the characteristics of the target domain. In this way, we achieve a highly harmonized source dataset for knowledge augmentation.

In summary, our contribution includes:

1. We introduce a novel automated data augmentation method, LEKA, which utilizes an automated retrieval method for external data and harmonizes it with target data for knowledge augmentation.
2. We develop a novel paradigm by incorporating an LLM into the data harmonization process to optimize data space and structure to enhance downstream machine learning performance.
3. We conduct a series of experiments to validate the effectiveness and robustness of our LEKA method across different tasks. Experimental results demonstrate that our method has clear advantages over existing methods.

2 Related Work

2.1 Knowledge Transfer

Knowledge transfer is a knowledge augmentation method that improves learning on new tasks by transferring knowledge from a related task [Alyafeai *et al.*, 2020; Wang *et al.*, 2022; Wang *et al.*, 2025]. Knowledge transfer allows cross-domain knowledge transformation despite data distribution or feature space differences. Specifically, knowledge transfer adapts models developed for one task to perform better on different but related tasks, as it adjusts the feature mappings and decision boundaries to suit new knowledge [Han *et al.*, 2021; Yordanov *et al.*, 2021]. Finding suitable pre-training knowledge is crucial for knowledge transfer because it significantly enhances the model's effectiveness by providing a relevant starting point. While many methods exist to adjust existing knowledge within the same or across different domains to improve transfer learning outcomes, the process of retrieving certain knowledge still heavily relies on manual effort.

2.2 Data Harmonizing with LLMs

Data harmonization applying LLMs offers significant benefits by leveraging their natural language capabilities to standardize and integrate diverse datasets and further enhance model performance through improved data consistency [Cao *et al.*, 2009; Zhang *et al.*, 2024a; Liu *et al.*, 2021; Durante *et al.*, 2024]. However, this approach has several challenges, including the high computational costs of training LLMs and potential biases in training data, which can adversely affect the harmonization process [Feng *et al.*, 2021; Xie *et al.*, 2024; Zhang *et al.*, 2025a]. Furthermore, the risk of overfitting remains a concern, as models may become overly specialized in training data nuances, reducing their effectiveness on new datasets.

2.3 Retrieval Augmented Generation for Knowledge Augmentation

Retrieval Augmented Generation (RAG) [Lewis *et al.*, 2020] for knowledge augmentation is a method that enhances the capabilities of generative language models by integrating information retrieval with model generation [Hu and Lu, 2024]. This technique aids LLMs in tasks that demand deep and specific domain knowledge [Zhang *et al.*, 2024b; Huang and Huang, 2024]. RAG for knowledge augmentation provides access to expansive external libraries (collections of documents or domain-relevant knowledge), making it especially effective for transferring knowledge across different domains [Siriwardhana *et al.*, 2023]. RAG for knowledge augmentation embeds extensive databases directly into the generative process so that specific, domain-related information is both accessible and effectively utilized. In this way, it

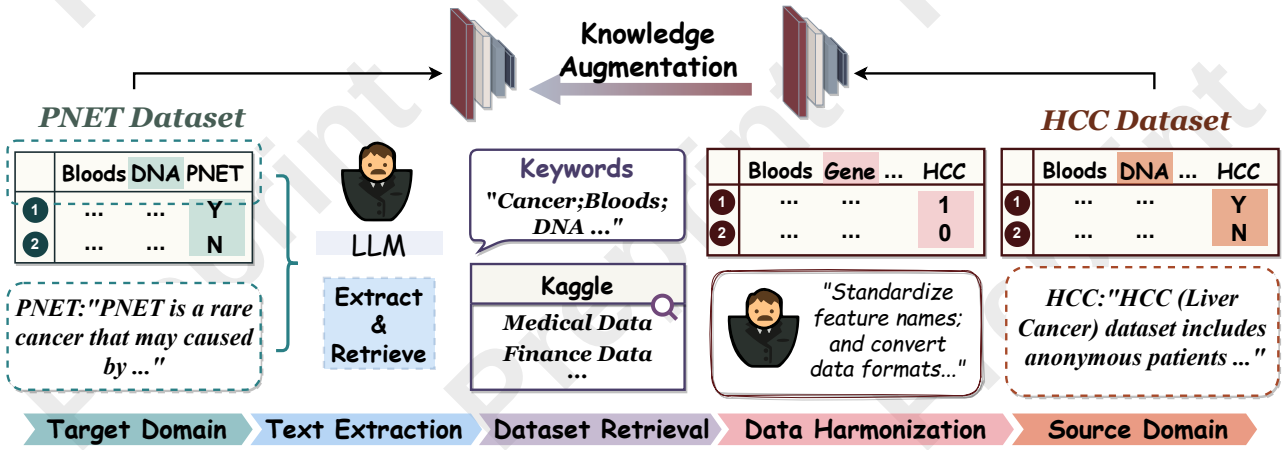

Figure 1: Example of LEKA. We adopt an LLM to retrieve proper source domain data to transfer knowledge to a data-limited target domain. The LLM extracts the key information of the target data to retrieve a relevant dataset; then, we adopt the LLM for harmonization.

can significantly enhance the performance for complex tasks in knowledge augmentation scenarios.

3 Preliminary

3.1 Definition

Definition 1. (Domain) A domain $\mathcal{D}$ is an ordered pair consisting of a feature space $\mathcal{X}$ and a marginal probability measure P defined on this feature space. In other words, $\mathcal{D} = (X, P)$, where $\mathbf{X} = \{\mathbf{x}|\mathbf{x}_i \in X, i = 1, \ldots, n\}$ is an instance set. And P is a probability measure that describes the probability of occurrence of feature vectors $\mathbf{x} \in \mathbf{X}$. This probability measure P makes $(\mathcal{X}, \mathcal{B}(\mathcal{X}), P)$ a probability space, where $\mathcal{B}(\mathcal{X})$ is the Borel σ-algebra on $\mathcal{X}$.

Source Domain. The source domain $\mathcal{D}_S$ consists of instances paired with labels y, represented as $\mathcal{D}_S = \{(\mathbf{x}, \mathbf{y})|\mathbf{x}_i \in \mathcal{X}_S, y_i \in \mathcal{Y}_S, i = 1, \ldots, n^S\}$. Here, $\mathcal{X}_S$ is the feature space and $\mathcal{Y}_S$ is the label space for the source domain. This domain provides labeled data used to train models in preparation for transfer learning tasks.

Target Domain. The target domain $\mathcal{D}_T$ typically contains a mix of unlabeled instances and a smaller set of labeled instances, denoted as $\mathcal{D}_T = \{\mathbf{x} \in \mathcal{X}_T\} \cup \{(\mathbf{x}, \mathbf{y})|(\mathbf{x}, \mathbf{y}) \in \mathcal{X}_T \times \mathcal{Y}_T\}$. Here, $\mathcal{X}_T$ is the feature space and $\mathcal{Y}_T$ is the label space for the target domain. We aim to evaluate and fine-tune the transfer learning models with data in the target domain.

Definition 2. (Task) A task $\mathcal{T}$ consists of a label space $\mathcal{Y}$ and a decision function f, formally noted as $\mathcal{T} = (\mathcal{Y}, f)$, where $\mathcal{Y}$ is a metric space that contains all possible labels, and f is a mapping from the feature space $\mathcal{X}$ to a set of conditional probability measures on the label space $\mathcal{Y}$.

Source Task. The learning task of the source task $\mathcal{T}_S$ is typically represented as learning a target function $f_S : \mathcal{X}_S \rightarrow \mathcal{Y}_S$, where $\mathcal{Y}_S$ is the label space of the source task.

Target Task. The learning task of the target task $\mathcal{T}_T$ is typically represented as learning a target function $f_T : \mathcal{X}_T \rightarrow \mathcal{Y}_T$, where $\mathcal{Y}_T$ is the label space of the target task.

Definition 3. (Knowledge Transfer) Knowledge transfer is an augmentation method that adopts observations from source domains and tasks, denoted as $\{(\mathcal{D}_{S_i}, \mathcal{T}_{S_i})|i = 1, \ldots, m^S\}$. Here $m^S \in \mathbb{N}^+$ represents the number of source domains and tasks. Similar observations from target domains and tasks are denoted as $\{(\mathcal{D}_{T_j}, \mathcal{T}_{T_j})|j = 1, \ldots, m^T\}$, where $m^T \in \mathbb{N}^+$. The goal of knowledge transfer is to utilize the knowledge embedded in the source domains $\mathcal{D}_{S_i}$ to enhance the performance of the learned decision functions f_{T_j} across the target domains $\mathcal{D}_{T_j}$, for $j = 1, \ldots, m^T$.

3.2 Problem Formulation

We formulate the task to enhance the performance of knowledge transfer through automated data retrieval and harmonization using an LLM. Concretely, we improve the performance of the learned decision function f_{T_j} by reconstructing and refining the source domain $\mathcal{D}_S$. In this way, we better utilize the knowledge implied in $\mathcal{D}_S$ as we mitigate domain shifts and facilitate a more effective transfer of learned models. Concretely, our optimization objective is to retrieve and reconstruct a source domain $\mathcal{D}_S^*$:

$$\mathcal{D}_S^* = \underset{\hat{\mathcal{D}}_S}{\operatorname{argmax}}\, \mathcal{P}_{\mathcal{D}_T}(f_{T_j}), \tag{1}$$

where $\mathcal{P}$ is the performance indicator of f_{T_j} and $\hat{\mathcal{D}}_S$ is a reconstructed source domain aligned with $\mathcal{D}_T$.

4 Methodology

In this section, we introduce our novel knowledge augmentation method named **L**LM-**E**nhanced **K**nowledge **A**ugmentation (**LEKA**), designed to dynamically and automatically retrieve and refine source data to transfer knowledge and further enhance target data learning. Specifically, our study focuses on tabular datasets as the concrete form of data. The LEKA leverages an LLM to extract essential information from the target dataset, including data structure, feature names, and descriptions. Within this framework, the LLM uses its capability to analyze and synthesize data to optimize the selection and refinement of source datasets. It extracts and summarizes keywords for retrieval tailored to the

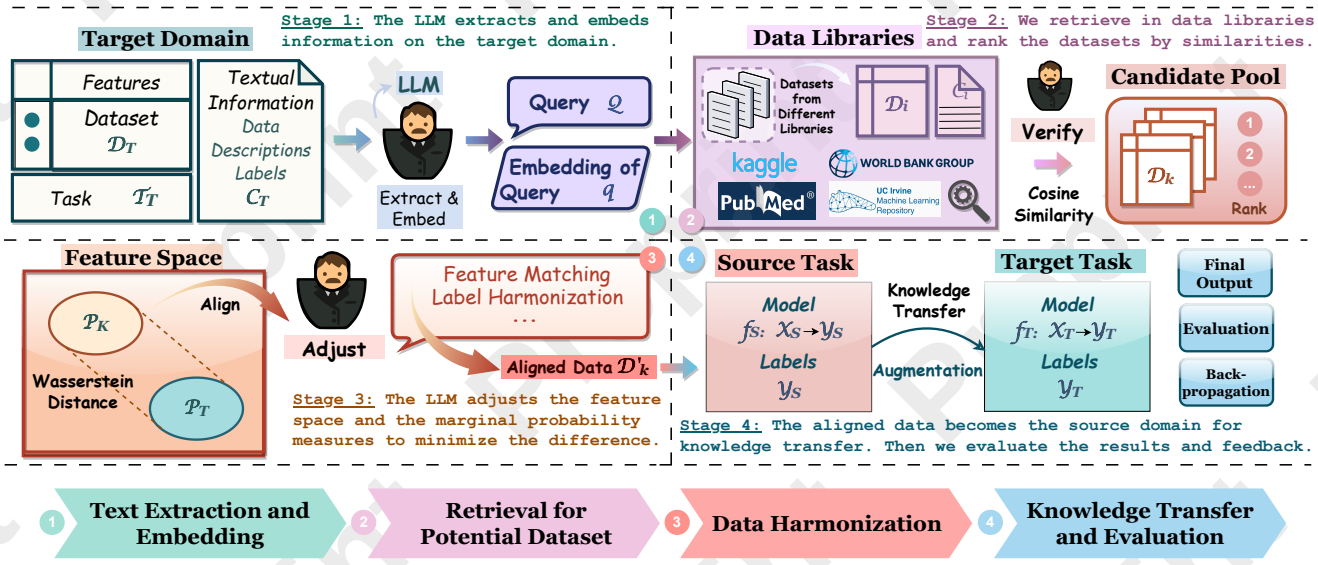

Figure 2: Framework of LEKA includes: 1) an LLM extracts and embeds the textual information of the target dataset, then 2) retrieves datasets in libraries, and 3) processes data harmonization. With harmonized datasets, we can transfer knowledge from the source dataset we construct to enhance learning on the target dataset.

target domain and task. After retrieval, the LLM refines and harmonizes these datasets in feature space and marginal probability measures with the target dataset. Figure 2 illustrates the overview of the LEKA framework, which comprises three stages: (1) **Dataset Retrieval**; (2) **Data Harmonization**; and (3) **Knowledge Transfer and Evaluation**.

4.1 Dataset Retrieval

In this phase, our primary objective is to identify and retrieve a source dataset $\mathcal{D}_S$ similar to the target dataset $\mathcal{D}_T$ for effective pre-training in knowledge transfer scenarios. To achieve this, we utilize an LLM $\mathcal{A}$ to analyze and retrieve data that matches the structural properties and purposes of $\mathcal{D}_T$. A straightforward method is to embed the target dataset $\mathcal{D}_T$ by $\mathcal{A}$ to capture its essential features. However, this approach can be computationally intensive and prone to errors. Inspired by the strategies outlined in [Zhang *et al.*, 2024c], we turn to leverage the textual information, including the data descriptions $\mathcal{C}_T$ and feature names l_T of $\mathcal{D}_T$. These textual elements provide insights into the structure and purpose of $\mathcal{D}_T$, and the LLM can precisely focus on the semantic content of $\mathcal{D}_T$.

First, the LLM $\mathcal{A}$ constructs a query $\mathcal{Q}$ based on this textual information:

$$\mathcal{Q} = \mathcal{A}(\mathcal{C}_T, l_T), \tag{2}$$

and then embeds the query:

$$q = \mathcal{A}(\text{embed}(\mathcal{Q})), \tag{3}$$

where $\text{embed}(\cdot)$ is the embedding process that transforms the input into a vector representation.

We then retrieve from a library $\mathcal{L}$ of datasets for the top-k most relevant datasets $\{\mathcal{D}_k\}$ to query $\mathcal{Q}$. We evaluate the relevance of all datasets in $\mathcal{L}$ to the query $\mathcal{Q}$ by the cosine similarity of the embeddings and select the top-k datasets with the highest similarity:

$$\text{sim}(q, d_k) = \frac{q \cdot d_k}{\|q\|\|d_k\|}, \tag{4}$$

where $d_k = \mathcal{A}(\text{embed}(\mathcal{D}_k))$ is the embedding of a potential dataset $\mathcal{D}_k$'s textual information, and $\|\cdot\| \in \mathbb{R}$ is the norm function.

In this phase, we map textual information to a high-dimensional vector space. Essentially, we approximate the probability distributions of the datasets in the library to retrieve k source datasets most similar in structure and purpose to the target dataset. We leverage the geometric properties of vectors in high-dimensional space to assess and quantify dataset similarities. This prepares us for further dataset processing for transfer learning.

4.2 Data Harmonization

Now that we have k potential source domain datasets, we turn to align these datasets with the target dataset $\mathcal{D}_T$ with an LLM. This alignment process adjusts the feature space and marginal probability measures of the source dataset $\mathcal{D}_k$ to closely match those of the target dataset $\mathcal{D}_T$. This process involves transformations of features and adjustments of distributions. Here we take $\mathcal{D}_k$ as an example.

Feature Space Transformation. We denote $\mathcal{X}_k$ and $\mathcal{X}_T$ as the feature spaces of the source dataset $\mathcal{D}_k$ and the target dataset $\mathcal{D}_T$, respectively. Our LLM constructs a mapping function $f : \mathcal{X}_k \to \mathcal{X}_T$ for alignment. This function transforms the features in $\mathcal{X}_k$ to minimize the distance between the transformed source features and the target features. Here, we aim to minimize the distance $d(f(\mathcal{X}_k), \mathcal{X}_T)$, where d is a kernel distance:

$$d(f(\mathcal{X}_k), \mathcal{X}_T) = \sqrt{\sum_{i,j} (\kappa(f(x_i^k), f(x_j^k)) - \kappa(x_i^T, x_j^T))^2}. \tag{5}$$

Here, $\kappa(\cdot, \cdot)$ is a kernel function, x_i^k and x_j^k are the i-th and j-th feature vectors in the source dataset $\mathcal{D}_k$. We adopt a Gaussian kernel $\kappa(x, y) = \exp(-\gamma \|x - y\|^2)$, with γ being a positive bandwidth parameter. This kernel function can effectively measure the similarity between points in high-dimensional spaces and captures both linear and non-linear relationships. Thus, it can handle the complexities inherent in high-dimensional data. Compared with Euclidean distances, kernel distances can capture the geometric structure of the data manifold. It provides a more robust and informative similarity measure for the source-target alignment.

Marginal Probability Measures Harmonization. Our LLM aligns the marginal probability measures $\mathcal{P}_k$ and $\mathcal{P}_T$ of the source and target datasets. The LLM analyzes and refines textual information such as labels, feature names, and classification probability distributions. To measure the differences between $\mathcal{D}_k$ and $\mathcal{D}_T$, we adopt the Wasserstein distance:

$$W(\mathcal{P}_k, \mathcal{P}_T) = \inf_{\gamma \in \Gamma(\mathcal{P}_k, \mathcal{P}_T)} \int_{\mathcal{X} \times \mathcal{X}} \|x - y\| \, d\gamma(x, y), \tag{6}$$

$$\mathcal{D}_k^{'} = \mathcal{A}(\mathcal{D}_k, \mathcal{P}_k, \mathcal{P}_T), \tag{7}$$

where $\Gamma(\mathcal{P}_k, \mathcal{P}_T)$ represents the set of all joint distributions with marginals $\mathcal{P}_k$ and $\mathcal{P}_T$ on $\mathcal{X} \times \mathcal{X}$. The Wasserstein distance minimizes the transportation cost while preserving the geometric structure of the data distributions. Specifically, the Wasserstein distance calculates the minimum "geographical" cost required to move data from one distribution to another, where "geographical" cost refers to the cost of moving data from one point to another in the feature space.

Moreover, the Wasserstein distance is advantageous when dealing with distributions whose support sets (i.e., the effective range or set of the distributions) do not fully overlap. In real-world data applications, it is common for the source and target datasets to originate from different distributions. Their data points do not align perfectly and cover the same areas. In these cases, traditional distance metrics like the Euclidean distance poorly reflect the actual differences between the two datasets, as they merely measure differences in position but ignore the overall structure of the data distributions.

With this source-target alignment process, the LLM improves the efficiency of knowledge transfer from the source domain to the target domain. Thus, it enhances the overall performance of our transfer learning models. Further, we optimize data handling and boost the adaptability and accuracy of transfer learning for adoption in various applications.

4.3 Knowledge Transfer and Evaluation

After aligning the source dataset $\mathcal{D}_k'$ with the target dataset $\mathcal{D}_T$, and adjusting the marginal probability measures $\mathcal{P}_k'$ to $\mathcal{P}_T$, we proceed to integrate these aligned datasets into the transfer learning model. At this stage, our goal is to transfer knowledge from $\mathcal{D}_k'$ to $\mathcal{D}_T$ by minimizing a defined loss function while adapting the model to the target domain.

To achieve this, the transfer learning process focuses on updating the decision function f_{T_j}. Specifically, we aim to minimize the expected loss over the target dataset while incorporating the knowledge transferred from the source dataset:

$$f_{T_j}^* = \arg\min_{f_{T_j}} \mathbb{E}_{(x,y) \in \mathcal{D}_T} [\mathcal{L}(f_{T_j}(x; \theta), y)], \tag{8}$$

where $\mathcal{L}$ is the loss function, x represents the features, y represents the labels in $\mathcal{D}_T$, and θ denotes the parameters of f_{T_j} that are being optimized. We calculate the expectation by the probability distribution $\mathcal{P}_T$, which has been closely aligned with $\mathcal{P}_k'$ to ensure consistency and maximize the efficacy of the knowledge transfer.

During update, the optimization of model parameters θ leverages both the target dataset $\mathcal{D}_T$ and the aligned source dataset $\mathcal{D}_k'$, incorporating domain-specific characteristics to reduce domain shift. We define the optimization process as:

$$\theta^* = \arg\min_{\theta} \Big(\alpha \mathbb{E}_{(x,y) \in \mathcal{D}_k'} [\mathcal{L}(f_{T_j}(x; \theta), y)] + (1 - \alpha) \mathbb{E}_{(x,y) \in \mathcal{D}_T} [\mathcal{L}(f_{T_j}(x; \theta), y)] \Big). \tag{9}$$

This formula aims to fine-tune the decision function f_{T_j} by minimizing the weighted sum of expected losses across the datasets. The loss function $\mathcal{L}(f_{T_j}(x; \theta), y)$ evaluates prediction accuracy, guiding the adjustment of parameters. The expectations $\mathbb{E}_{(x,y) \in \mathcal{D}_k'}$ and $\mathbb{E}_{(x,y) \in \mathcal{D}_T}$ represent the mean losses over the source and target datasets, respectively. The weighting factor α adjusts the relative influence of each dataset and enables flexible adaptation between leveraging established knowledge from the source and integrating new data from the target. In this way, we apply the transferred knowledge to enhance model performance on the target task.

Following the optimization of the model parameters, the backpropagation process then updates θ by minimizing the total loss. This loss is a weighted sum calculated from the losses on $\mathcal{D}_k'$ and $\mathcal{D}_T$. This involves calculating the gradient of the loss function with respect to θ and updating θ using gradient descent methods:

$$\theta \leftarrow \theta - \eta \nabla_\theta \left(\alpha \mathcal{L}(\mathcal{D}_k'; \theta) + (1 - \alpha) \mathcal{L}(\mathcal{D}_T; \theta) \right), \tag{10}$$

where η is the learning rate. The gradients are computed based on both datasets, which allows the model to learn from both the aligned source data and the target data.

After the parameter optimization and backpropagation, we evaluate the effectiveness of the transfer learning process; we adopt a performance metric ϕ on the target dataset:

$$\phi(f_{T_j}^*, \mathcal{D}_T), \tag{11}$$

where ϕ measures the performance of the optimized model $f_{T_j}^*$ on $\mathcal{D}_T$.

In this way, we systematically improve the model based on empirical performance metrics.

5 Experiments

In this section, we present four experiments to demonstrate the effectiveness and impacts of the LEKA. First, we compare

Datasets	Samples	Features	Class
BCW	570	30	2
VID	8631	22	50
HD	303	13	2
TCC	7043	21	2

Table 1: Datasets description. Here we use four datasets from the medical and economic domains.

Dataset	Metrics	FTT	TTab	LEKA
BCW	Acc	0.956	0.956	**0.991**
	Prec	0.951	0.948	**0.988**
	Rec	0.960	0.960	**0.993**
	F1	0.955	0.954	**0.990**
VID	Acc	0.745	0.797	**0.995**
	Prec	0.758	0.588	**0.996**
	Rec	0.747	0.513	**0.996**
	F1	0.739	0.519	**0.996**
HD	Acc	0.738	0.803	**0.918**
	Prec	0.726	0.802	**0.914**
	Rec	0.718	0.802	**0.918**
	F1	0.721	0.802	**0.916**
TCC	Acc	0.836	0.795	**0.887**
	Prec	0.803	0.738	**0.846**
	Rec	0.865	0.712	**0.901**
	F1	0.817	0.722	**0.865**

Table 2: Performance comparison of transfer learning methods.

the LEKA against several baseline methods on four downstream tasks. Second, we present the correlations between several target domains and their retrieved source domains. Finally, we discuss the reason for performance improvement.

5.1 Experiment Settings

Datasets and Domains. We evaluate our method on four datasets of medical and economic domains: (1) ***Breast Cancer Wisconsin*** *(Diagnostic) (BCW)* [Wolberg *et al.*, 1995], (2) ***Heart Disease*** *(HD)* [Janosi *et al.*, 1989], (3) ***Vehicle Insurance Data*** *(VID)* [Bhatt, 2019], and (4) ***Telco Customer Churn*** *(TCC)* [BlastChar, 2018]. We show the detailed information about the features of the datasets in Table 1.

Metrics and Models. We evaluate the model performance by the following metrics: *Overall Accuracy (Acc)* measures the proportion of true results (both true positives and true negatives) in the total dataset. *Precision (Prec)* reflects the ratio of true positive predictions to all positive predictions for each class. *Recall (Rec)*, also known as sensitivity, reflects the ratio of true positive predictions to all actual positives for each class. *F-Measure (F1)* is the harmonic mean of precision and recall, calculated here as the macro-average. We apply the LEKA across a range of models: 1) *Tabnet (TN)* [Arik and Pfister, 2021]; 2) *TabTransformer (TT)* [Huang *et al.*, 2020]; 3) *Random Forest (RF)* [Rigatti, 2017]; 4) *Gradient Boosting Decision Trees (GBDT)* [Lin *et al.*, 2023] ; 5) *XGBoost (XB)* [Chen and Guestrin, 2016]. We compare the performance in these tasks both with and without our method.

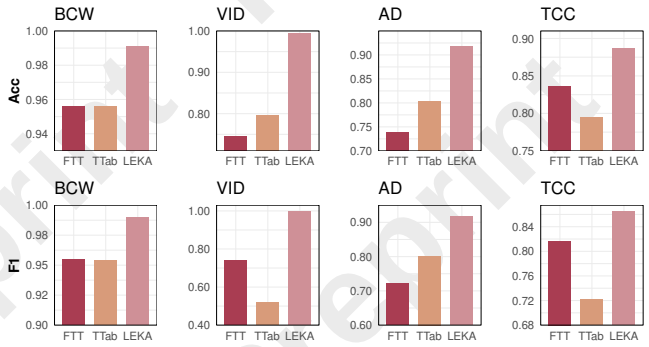

Figure 3: Comparison of accuracy and F1 scores on various transfer learning methods.

Baseline Models. We compare the LEKA with six baseline methods, including: 1) **Raw Data**: using vanilla data; 2) **TIFG** [Zhang *et al.*, 2024c]: feature generation with LLM; 3) **KPDDS** [Huang *et al.*, 2024]: data synthesis with LLM using key point examples; 4) **GReaT** [Borisov *et al.*, 2022]: data synthesis with LLM simulating data subset distributions; 5) **FTT** [Levin *et al.*, 2022]: A TabTransformer pretrained on the source domain and then fine-tuned on the target domain; 6) **TTab** [Wang and Sun, 2022]: A transferable tabular Transformer capable of learning from multiple tabular datasets.

Implementation Details. In our setup for data synthesis and model training, we utilize `GPT-4o` [OpenAI, 2024] as the query generator, combined with the `Exa API` [Exa, 2024] to fetch web pages containing datasets from `Kaggle` [Kaggle, 2024] and the `UCI Machine Learning Repository` [University of California, Irvine, 2024] that may be suitable for knowledge transfer. We extract dataset descriptions from web pages, and `GPT-4o` assesses their potential for knowledge transfer. Additionally, `GPT-4o` serves as a generator for executable code, performing up to five code generations. For our models, we configure TN, TT, and FTT with a batch size of 512 for the VID and TCC datasets and a batch size of 32 for the BCW dataset, a maximum of 100 epochs, and employ early stopping with a patience of 20. The learning rate is set at the default 0.02 for `pytorch_tabnet`. For the RF and GBDT models, the number of trees is set to 100, with GBDT also configured with a learning rate of 0.1 and a max depth of 3. TTab is set with a maximum of 50 epochs, a learning rate of 1×10^{-3}, and a weight decay of 1×10^{-4}.

5.2 Experiment Results

Overall Performance. The results are shown in Table 3, comparing our LEKA across four datasets. In summary:

(1) Compared with baseline models, LEKA significantly improves accuracy, surpassing baseline methods like TIFG and GReaT, with enhancements up to 4.4% in models like GBDT. LEKA consistently achieves top results in precision and recall, indicating its precision in correctly identifying relevant cases while minimizing false positives. The F1 scores under LEKA are notably high, reflecting its effective bal-

Metrics	Model	GBDT				RF				TN				XB				TT			
		BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC
Acc	Raw	0.939	0.991	0.754	0.793	0.921	0.827	0.770	0.776	0.956	0.436	0.803	0.783	0.930	0.946	0.770	0.773	0.965	0.714	0.574	0.754
	KPDDS	0.947	0.992	0.787	0.787	0.939	0.846	0.803	0.787	0.982	0.567	0.820	0.806	0.956	0.948	0.836	0.789	0.974	0.728	0.639	0.781
	GReaT	0.974	0.994	0.803	0.808	0.947	0.847	0.852	0.788	0.965	0.506	0.885	0.795	0.974	0.953	0.820	0.784	0.974	0.724	0.656	0.754
	TIFG	0.947	0.994	0.836	0.815	0.947	0.858	0.869	0.800	0.982	0.579	0.902	0.815	0.965	0.952	0.852	0.803	0.965	0.736	0.705	0.790
	LEKA	**0.991**	**0.995**	**0.869**	**0.825**	**0.982**	**0.868**	**0.885**	**0.811**	**0.982**	**0.596**	**0.918**	**0.838**	**0.991**	**0.960**	**0.885**	**0.823**	**0.991**	**0.787**	**0.754**	**0.887**
		BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC
Prec	Raw	0.948	0.992	0.804	0.734	0.926	0.834	0.751	0.716	0.949	0.429	0.803	0.717	0.928	0.932	0.775	0.706	0.965	0.742	0.302	0.709
	KPDDS	0.936	0.993	0.782	0.720	0.929	0.850	0.807	0.725	0.978	0.594	0.823	0.767	0.954	0.934	0.843	0.744	0.974	0.752	**0.817**	0.724
	GReaT	0.974	0.994	0.804	0.752	0.949	0.851	0.852	0.743	0.959	0.448	0.895	0.746	0.966	0.955	0.823	0.748	0.979	0.732	0.659	0.697
	TIFG	0.947	0.994	0.840	0.767	0.940	0.863	0.867	0.750	0.981	0.571	0.900	0.748	0.967	0.957	0.849	0.750	0.965	0.725	0.707	0.734
	LEKA	**0.988**	**0.996**	**0.863**	**0.782**	**0.980**	**0.869**	**0.887**	**0.760**	**0.977**	**0.619**	**0.914**	**0.823**	**0.989**	**0.962**	**0.885**	**0.785**	**0.991**	**0.802**	0.752	**0.846**
		BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC
Rec	Raw	0.912	0.991	0.770	0.701	0.918	0.828	0.751	0.671	0.954	0.443	0.803	0.680	0.928	0.926	0.769	0.682	0.965	0.720	0.461	0.757
	KPDDS	0.959	0.993	0.785	0.689	0.951	0.845	0.806	0.675	0.986	0.553	0.819	0.686	0.962	0.930	0.845	0.710	0.974	0.734	0.522	0.755
	GReaT	0.968	0.994	0.804	0.717	0.944	0.846	0.853	0.702	0.967	0.502	0.887	0.692	0.973	0.954	0.827	0.704	0.968	0.726	0.661	0.717
	TIFG	0.935	0.994	0.839	0.725	0.940	0.856	0.867	0.697	0.981	0.570	0.904	0.700	0.957	0.951	0.852	0.711	0.965	0.733	0.690	0.724
	LEKA	**0.993**	**0.996**	**0.877**	**0.740**	**0.985**	**0.865**	**0.887**	**0.725**	**0.986**	**0.603**	**0.918**	**0.732**	**0.993**	**0.958**	**0.885**	**0.747**	**0.991**	**0.789**	**0.768**	**0.901**
		BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC	BCW	VID	HD	TCC
F1	Raw	0.927	0.991	0.750	0.714	0.920	0.827	0.751	0.685	0.952	0.400	0.803	0.693	0.928	0.927	0.769	0.692	0.965	0.704	0.365	0.718
	KPDDS	0.944	0.993	0.783	0.701	0.936	0.844	0.803	0.691	0.982	0.510	0.819	0.708	0.956	0.930	0.836	0.723	0.974	0.724	0.429	0.735
	GReaT	0.971	0.994	0.803	0.731	0.946	0.846	0.852	0.716	0.963	0.440	0.885	0.709	0.970	0.952	0.819	0.718	0.973	0.712	0.655	0.704
	TIFG	0.941	0.994	0.836	0.741	0.940	0.853	0.867	0.715	0.981	0.535	0.901	0.718	0.961	0.951	0.850	0.726	0.965	0.706	0.691	0.729
	LEKA	**0.990**	**0.996**	**0.866**	**0.756**	**0.982**	**0.864**	**0.885**	**0.739**	**0.981**	**0.568**	**0.916**	**0.760**	**0.991**	**0.959**	**0.885**	**0.761**	**0.991**	**0.780**	**0.750**	**0.865**

Table 3: Overall performance on downstream tasks. The best results are highlighted in **bold**, and the runner-up results are highlighted in underline. (Higher values indicate better performance.)

ance between precision and recall across various models and datasets. These results validate the effectiveness of LEKA 's retrieval and harmonization strategies and its robustness in diverse application scenarios. Overall, LEKA's strategic approach to knowledge transfer is particularly advantageous in complex data environments, showcasing its adaptability and efficiency compared to traditional methods.

(2) The LEKA outperforms all baseline methods in almost all metrics and datasets. Specifically, LEKA shows enhancements in accuracy by $2 - 5\%$ over other methods. It demonstrates notable improvements of up to 4.4% in GBDT for the VID dataset and 5.4% in RF for the BCW dataset. For improvement of overall accuracy, we demonstrate the LEKA's effective retrieval and harmonization of feature space and marginal probability measures with the target datasets. LEKA demonstrates exceptional precision, improving the TT metric for the TCC dataset by over 10% compared to baselines, effectively reducing false positives. Meanwhile, the LEKA consistently outperforms baselines in reducing misclassification rates with the highest F1 scores. These results prove that LEKA's retrieval and data harmonization reduce misclassification by forming a deeper and clearer understanding of the potential relationship between features.

Comparison with transfer learning methods. We then compare our LEKA method with transfer learning methods to demonstrate its effectiveness in enhancing model performance in complex domains. In these scenarios, the transfer learning methods show their reliance on manual data selection and alignment processes. The results demonstrate that LEKA outperforms transfer learning methods, FFT, and TTab across various metrics and datasets. For example, in the BCW dataset, LEKA improves accuracy by 3.5% and recall by 3.3% compared to the competing methods. This superior performance is consistent across other datasets like VID, HD, and TCC, with notable enhancements in precision and recall, emphasizing LEKA's effective data harmonization capabilities. This adaptability and efficiency in handling diverse datasets underscore LEKA's robustness and advantages.

6 Conclusion

In this paper, we introduce LLM-Enhanced Knowledge Augmentation (LEKA), a novel retrieval and harmonization framework that dynamically refines source data for effective knowledge transfer. This structure significantly enhances data augmentation by leveraging advanced LLM capabilities to automatically align and optimize data retrieved from diverse external libraries. Extensive experiments across various tasks demonstrate superiority to existing methods, especially in improving model adaptability and accuracy in complex data environments. By adopting LEKA on data-scarce domains, we achieve substantial improvements in learning performance and domain-specific task accuracy. For future work, we plan to extend this framework to include more varied data types and conduct a thorough empirical analysis to understand its underlying mechanisms and impacts.

7 Limitations

We acknowledge the following limitations: (1) the current work has only been tested on tabular data, and more complex test scenarios have not yet been involved; (2) despite its automation, LEKA can be computationally intensive. This could limit its applicability in resource-constrained environments or require substantial computational resources to maintain operational efficiency; (3) adopting the LEKA in tasks with unique requirements may have inherent limitations.

Acknowledgements

Dr. Kunpeng Liu is supported by the National Science Foundation (NSF) via the grant numbers 2426339 and 2348485. Dr. Yanjie Fu is supported by the National Science Foundation (NSF) via the grant numbers 2426340, 2416727, 2421864, 2421865, 2421803, and National Academy of Engineering Grainger Foundation Frontiers of Engineering Grants.

References

[Alyafeai *et al.*, 2020] Zaid Alyafeai, Maged Saeed AlShaibani, and Irfan Ahmad. A survey on transfer learning in natural language processing. *arXiv preprint arXiv:2007.04239*, 2020.

[Arik and Pfister, 2021] Sercan Ö Arik and Tomas Pfister. Tabnet: Attentive interpretable tabular learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35(8), pages 6679–6687, 2021.

[Bhatt, 2019] Himanshu Bhatt. Vehicle Insurance Data. https://www.kaggle.com/datasets/junglisher/vehicle-insurance-data, 2019. Accessed: 2025-01-21.

[BlastChar, 2018] BlastChar. Telco Customer Churn. https://www.kaggle.com/datasets/blastchar/telco-customer-churn, 2018. Accessed: 2025-01-21.

[Borisov *et al.*, 2022] Vadim Borisov, Kathrin Seßler, Tobias Leemann, Martin Pawelczyk, and Gjergji Kasneci. Language models are realistic tabular data generators. *arXiv preprint arXiv:2210.06280*, 2022.

[Cao *et al.*, 2009] Longbing Cao, Vladimir Gorodetsky, and Pericles A Mitkas. Agent mining: The synergy of agents and data mining. *IEEE intelligent systems*, 24(3):64–72, 2009.

[Chen and Guestrin, 2016] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[Durante *et al.*, 2024] Zane Durante, Qiuyuan Huang, Naoki Wake, Ran Gong, Jae Sung Park, Bidipta Sarkar, Rohan Taori, Yusuke Noda, Demetri Terzopoulos, Yejin Choi, et al. Agent ai: Surveying the horizons of multimodal interaction. *arXiv preprint arXiv:2401.03568*, 2024.

[Edge *et al.*, 2024] Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*, 2024.

[Exa, 2024] Exa. Exa API. https://exa.ai/, 2024. Accessed: 2025-01-10.

[Feng *et al.*, 2021] Steven Y Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. A survey of data augmentation approaches for nlp. *arXiv preprint arXiv:2105.03075*, 2021.

[Fleischer *et al.*, 2024] Daniel Fleischer, Moshe Berchansky, Moshe Wasserblat, and Peter Izsak. Rag foundry: A framework for enhancing llms for retrieval augmented generation. *arXiv preprint arXiv:2408.02545*, 2024.

[Gao *et al.*, 2023] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2023.

[Guo *et al.*, 2017] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. Deepfm: a factorization-machine based neural network for ctr prediction. *arXiv preprint arXiv:1703.04247*, 2017.

[Han *et al.*, 2021] Wenjuan Han, Bo Pang, and Yingnian Wu. Robust transfer learning with pretrained language models through adapters. *arXiv preprint arXiv:2108.02340*, 2021.

[Hu and Lu, 2024] Yucheng Hu and Yuxing Lu. Rag and rau: A survey on retrieval-augmented language model in natural language processing. *arXiv preprint arXiv:2404.19543*, 2024.

[Huang and Huang, 2024] Yizheng Huang and Jimmy Huang. A survey on retrieval-augmented text generation for large language models. *arXiv preprint arXiv:2404.10981*, 2024.

[Huang *et al.*, 2020] Xin Huang, Ashish Khetan, Milan Cvitkovic, and Zohar Karnin. Tabtransformer: Tabular data modeling using contextual embeddings. *arXiv preprint arXiv:2012.06678*, 2020.

[Huang *et al.*, 2024] Yiming Huang, Xiao Liu, Yeyun Gong, Zhibin Gou, Yelong Shen, Nan Duan, and Weizhu Chen. Key-point-driven data synthesis with its enhancement on mathematical reasoning. *arXiv preprint arXiv:2403.02333*, 2024.

[Janosi *et al.*, 1989] Andras Janosi, William Steinbrunn, Matthias Pfisterer, and Robert Detrano. Heart Disease. https://archive.ics.uci.edu/dataset/45/heart+disease, 1989. Accessed: 2025-01-21.

[Jin *et al.*, 2024] Chao Jin, Zili Zhang, Xuanlin Jiang, Fangyue Liu, Xin Liu, Xuanzhe Liu, and Xin Jin. Ragcache: Efficient knowledge caching for retrieval-augmented generation. *arXiv preprint arXiv:2404.12457*, 2024.

[Kaggle, 2024] Kaggle. Kaggle: Your Machine Learning and Data Science Community. https://www.kaggle.com/, 2024. Accessed: 2025-01-10.

[Khodaee *et al.*, 2024] Pouya Khodaee, Herna L Viktor, and Wojtek Michalowski. Knowledge transfer in lifelong machine learning: a systematic literature review. *Artificial Intelligence Review*, 57(8):217, 2024.

[Levin *et al.*, 2022] Roman Levin, Valeriia Cherepanova, Avi Schwarzschild, Arpit Bansal, C Bayan Bruss, Tom Goldstein, Andrew Gordon Wilson, and Micah Goldblum. Transfer learning with deep tabular models. *arXiv preprint arXiv:2206.15306*, 2022.

[Lewis *et al.*, 2020] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

[Li and Ramakrishnan, 2025] Sha Li and Naren Ramakrishnan. Oreo: A plug-in context reconstructor to enhance retrieval-augmented generation. *arXiv preprint arXiv:2502.13019*, 2025.

[Lin *et al.*, 2023] Huawei Lin, Jun Woo Chung, Yingjie Lao, and Weijie Zhao. Machine unlearning in gradient boosting decision trees. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1374–1383, 2023.

[Liu *et al.*, 2021] Kunpeng Liu, Yanjie Fu, Le Wu, Xiaolin Li, Charu Aggarwal, and Hui Xiong. Automated feature selection: A reinforcement learning perspective. *IEEE Transactions on Knowledge and Data Engineering*, 35(3):2272–2284, 2021.

[Nam *et al.*, 2024] Jaehyun Nam, Woomin Song, Seong Hyeon Park, Jihoon Tack, Sukmin Yun, Jaehyung Kim, Kyu Hwan Oh, and Jinwoo Shin. Tabular transfer learning via prompting llms. *arXiv preprint arXiv:2408.11063*, 2024.

[OpenAI, 2024] OpenAI. GPT-4o. https://platform.openai.com/docs/models/gpt-4o, 2024. Accessed: 2025-01-10.

[Rigatti, 2017] Steven J Rigatti. Random forest. *Journal of Insurance Medicine*, 47(1):31–39, 2017.

[Ringwald and Stiefelhagen, 2021] Tobias Ringwald and Rainer Stiefelhagen. Adaptiope: A modern benchmark for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 101–110, 2021.

[Seemakhupt *et al.*, 2024] Korakit Seemakhupt, Sihang Liu, and Samira Khan. Edgerag: Online-indexed rag for edge devices. *arXiv preprint arXiv:2412.21023*, 2024.

[Shi *et al.*, 2018] Hongtao Shi, Hongping Li, Dan Zhang, Chaqiu Cheng, and Xuanxuan Cao. An efficient feature generation approach based on deep learning and feature selection techniques for traffic classification. *Computer Networks*, 132:81–98, 2018.

[Siriwardhana *et al.*, 2023] Shamane Siriwardhana, Rivindu Weerasekera, Elliott Wen, Tharindu Kaluarachchi, Rajib Rana, and Suranga Nanayakkara. Improving the domain adaptation of retrieval augmented generation (rag) models for open domain question answering. *Transactions of the Association for Computational Linguistics*, 11:1–17, 2023.

[Tang *et al.*, 2020] Zhiqiang Tang, Yunhe Gao, Leonid Karlinsky, Prasanna Sattigeri, Rogerio Feris, and Dimitris Metaxas. Onlineaugment: Online data augmentation with less domain knowledge. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 313–329. Springer, 2020.

[University of California, Irvine, 2024] University of California, Irvine. UCI Machine Learning Repository. https://archive.ics.uci.edu/, 2024. Accessed: 2025-01-10.

[Wang and Sun, 2022] Zifeng Wang and Jimeng Sun. Transtab: Learning transferable tabular transformers across tables. *Advances in Neural Information Processing Systems*, 35:2902–2915, 2022.

[Wang *et al.*, 2022] Limin Wang, Xinhao Zhang, Kuo Li, and Shuai Zhang. Semi-supervised learning for k-dependence bayesian classifiers. *Applied Intelligence*, pages 1–19, 2022.

[Wang *et al.*, 2025] Zaitian Wang, Jinghan Zhang, Xinhao Zhang, Kunpeng Liu, Pengfei Wang, and Yuanchun Zhou. Diversity-oriented data augmentation with large language models. *arXiv preprint arXiv:2502.11671*, 2025.

[Wolberg *et al.*, 1995] William Wolberg, Olvi Mangasarian, Nick Street, and W. Street. Breast Cancer Wisconsin (Diagnostic). https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic, 1995. Accessed: 2025-01-21.

[Xie *et al.*, 2024] Henry J Xie, Jinghan Zhang, Xinhao Zhang, and Kunpeng Liu. Scoring with large language models: A study on measuring empathy of responses in dialogues. *arXiv preprint arXiv:2412.20264*, 2024.

[Yordanov *et al.*, 2021] Yordan Yordanov, Vid Kocijan, Thomas Lukasiewicz, and Oana-Maria Camburu. Few-shot out-of-domain transfer learning of natural language explanations in a label-abundant setup. *arXiv preprint arXiv:2112.06204*, 2021.

[Zhang *et al.*, 2024a] Jinghan Zhang, Xiting Wang, Yiqiao Jin, Changyu Chen, Xinhao Zhang, and Kunpeng Liu. Prototypical reward network for data-efficient rlhf. *arXiv preprint arXiv:2406.06606*, 2024.

[Zhang *et al.*, 2024b] Tianjun Zhang, Shishir G Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E Gonzalez. Raft: Adapting language model to domain specific rag. *arXiv preprint arXiv:2403.10131*, 2024.

[Zhang *et al.*, 2024c] Xinhao Zhang, Jinghan Zhang, Fengran Mo, Yuzhong Chen, and Kunpeng Liu. Tifg: Text-informed feature generation with large language models. *arXiv preprint arXiv:2406.11177*, 2024.

[Zhang *et al.*, 2024d] Xinhao Zhang, Jinghan Zhang, Banafsheh Rekabdar, Yuanchun Zhou, Pengfei Wang, and Kunpeng Liu. Dynamic and adaptive feature generation with llm. *arXiv preprint arXiv:2406.03505*, 2024.

[Zhang *et al.*, 2025a] Jinghan Zhang, Fengran Mo, Xiting Wang, and Kunpeng Liu. Blind spot navigation in llm reasoning with thought space explorer, 2025.

[Zhang *et al.*, 2025b] Jinghan Zhang, Xiting Wang, Weijieying Ren, Lu Jiang, Dongjie Wang, and Kunpeng Liu. Ratt: A thought structure for coherent and correct llm reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39(25), pages 26733–26741, 2025.