SAGE: Sparse Adaptive Guidance for Dependency-Aware Tabular Data Generation

Shuo Yang* **Zheyu Zhang*** **Bardh Prenkaj** **Gjergji Kasneci**

Technical University of Munich
Munich Center for Machine Learning (MCML)
{name.surname}@tum.de

Abstract

Generating high-fidelity synthetic tabular data remains a critical challenge for enhancing data availability in privacy-sensitive and low-resource domains. Recent approaches leverage LLMs by representing table rows as sequences, yet suffer from two fundamental limitations: (1) they model feature dependencies densely, introducing spurious correlations; and (2) they assume static relationships between features, ignoring how these dependencies vary with feature values. To overcome these limitations, we introduce SAGE (Sparse Adaptive Guidance), a novel LLM-based generation framework that enforces sparse and dynamic dependency guidance. SAGE discretizes features into value-aware pseudo-features and constructs a mutual information-based sparse dependency graph. This graph adaptively guides generation through explicit context selection or implicit logit correction, enabling LLMs to focus on truly relevant information during synthesis. Our extensive experiments across six datasets and multiple tasks reveal that SAGE not only improves data fidelity and downstream utility, boosting F1 scores by 10% compared to previous LLM-based methods, but also reduces policy violations by one point. These results highlight the importance of adaptive structure in tabular data generation and provide new insights into context-sensitive control of LLMs.[1]

1 Introduction

Tabular data forms the backbone of decision-making across healthcare (Vallevik et al., 2024), finance (Sattarov et al., 2023), and education (Qu et al., 2022), yet obtaining high-quality datasets remains challenging due to privacy constraints and data collection costs (Borisov et al., 2022). This scarcity has driven significant interest in synthetic tabular data generation, which promises to unlock data-driven applications while preserving privacy (Liu et al., 2024; Zhao et al., 2025).

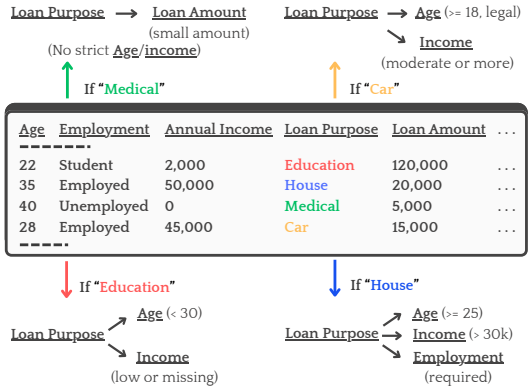

Figure 1: Value-conditioned dynamic dependencies in tabular data. Dependencies such as *Loan Purpose* → *Age, Income* vary with purpose: "Education" loans imply youth and low income, while "House" loans require stable employment and higher income. Static models overlook such conditional structures.

Early approaches have primarily focused on learning underlying data distributions through neural generative models (Stoian et al., 2025). Traditional methods based on variational autoencoders (Xu et al., 2019) and generative adversarial networks (Kamthe et al., 2021) learn tabular value distributions and generate new records by sampling from the latent space. However, these methods often produce logically inconsistent records, such as pairing professional occupations with minor ages (Yang et al., 2024; Long et al., 2025).

Recent work has addressed this limitation by leveraging large language models (LLMs), which benefit from strong sequential modeling capabilities and rich pre-trained knowledge (Liu et al., 2023a; Zhang et al., 2023; Han et al., 2025). These approaches convert tabular rows to textual sequences using templates like "feature is value" (Borisov et al., 2023), naturally incorporating world knowledge and dramatically reducing

*Equal contribution.
[1]Our code is publicly available at https://github.com/ShuoYangtum/SAGE.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 3792–3807
July 2-7, 2026

implausible outputs. However, current LLM-based generators face two fundamental limitations. First, they model feature relationships densely through fully-connected attention mechanisms, despite tabular data being inherently sparse (Liu et al., 2023b), with dependencies present only among limited subsets of features. This introduces spurious correlations and computational overhead. Second, they assume static dependencies between features, failing to capture how relationships change with specific values. Figure 1 illustrates this dynamic behavior: "Education" loans correlate with young applicants and low income, while "House" loans require stable employment and higher income thresholds. Existing dependency-aware methods (Xu et al., 2024) rely on pre-annotated, fixed dependency graphs that cannot adapt to such value-conditioned patterns.

As a two-pronged solution, we propose SAGE (Sparse Adaptive Guidance), a novel framework that enforces both sparse and dynamic dependency modeling for LLM-based tabular data generation. SAGE discretizes features into value-aware pseudo-features and constructs mutual information-based sparse dependency graphs that dynamically adapt during generation. This enables the model to focus on truly relevant feature relationships while filtering out spurious correlations. We propose two guidance strategies: (1) explicit context selection through *Feature Selector* and (2) implicit adjustment via *Logit Correction*, both designed to ensure that LLMs condition on contextually appropriate information when generating each feature value.

Our contributions are summarized as follows:

1. We propose SAGE, a novel tabular data synthesis framework that jointly models sparse feature dependencies and their dynamic variations based on feature values. Unlike existing methods that assume static relationships, SAGE adapts dependency structures during generation through mutual information-guided pseudo-feature discretization.
2. We introduce two complementary guidance mechanisms: *Feature Selector* for explicit context filtering and *Logit Correction* for implicit confidence adjustment. Additionally, we present engineering optimizations including value-only loss computation and rejection sampling to address computational overhead and invalid value generation in LLM-based approaches.
3. We conduct comprehensive experiments across six diverse datasets, evaluating downstream utility, data fidelity, privacy preservation, and realism. SAGE consistently outperforms existing methods, achieving up to 10% F1 improvement on classification tasks and reducing policy violation rates by over 1 percentage point compared to state-of-the-art LLM-based generators.

2 Related Work

End-to-End Tabular Generative Models. Tabular data generation methods rely on end-to-end neural architectures to capture the full joint distribution (Hollmann et al., 2025). Unlike vision or NLP domains, tabular datasets are often limited in size, making synthetic data generation particularly valuable for data augmentation (Shi et al., 2025). Xu et al. (2019) pioneered this field with CTGAN and TVAE, addressing imbalanced categorical features through adversarial training and providing stable probabilistic generation via variational autoencoders. Subsequent GAN-based methods (Zhang et al., 2021; Kim et al., 2021; Zhao et al., 2021, 2024) refined adversarial training but often suffer from mode collapse and poor interpretability. More recently, diffusion-based models such as TabDDPM (Kotelnikov et al., 2023), FinDiff (Sattarov et al., 2023), AutoDiff (Suh et al., 2023), and TabSyn (Zhang et al., 2024) emerged as promising alternatives through iterative denoising processes and architectural improvements. However, these models treat data as value matrices, largely ignoring the semantic meaning of features.

Language Models for Tabular Data Modeling. Motivated by the impressive performance of large language models, recent research explores tabular data generation by representing table rows as sequences of feature-value pairs. GReaT (Borisov et al., 2023) first demonstrated this approach through autoregressive language modeling, effectively leveraging pretrained world knowledge. Subsequent works like Pred-LLM (Nguyen et al., 2024) and TabuLa (Zhao et al., 2025) optimized feature-value representations to enhance correlation modeling, while P-TA (Yang et al., 2024) used proximal policy optimization to integrate GAN-based discriminator feedback. Prompt-based methods have emerged as an alternative paradigm: EPIC (Kim et al., 2024) and TabGen-ICL (Fang et al., 2025) demonstrated effective in-context learning for tabular synthesis, and CLLM (Seedat et al., 2024) leveraged LLM prior knowledge for data augmentation in low-data regimes. While these meth-

ods effectively incorporate feature semantics, they overlook the structured nature of feature dependencies by modeling entire rows as flat sequences, leading to high sampling latency.

Dependency Modeling in Tabular Data. To address the limitations of black-box models and flat-sequence representations, another line of research explicitly models the sparse, structured dependencies in tabular data. Early approaches integrated structural priors into classical generative frameworks: GOGGLE (Liu et al., 2023b) encodes pairwise feature relationships into graphs within a VAE, GANBLR (Zhang et al., 2021) incorporates auxiliary Bayesian Networks into a GAN, and DRL (Stoian and Giunchiglia, 2025) imposes logical rules through differentiable layers compatible with gradient-based training. More recently, structure-aware LLM-based methods constrain generation to follow predefined structures. SPADA (Yang et al., 2025) induces a sparse dependency graph that dictates the generation process, PAFT (Xu et al., 2024) aligns generation with a statistically-determined feature order, and GRADE (Zhang et al., 2025) uses statistical dependencies to dynamically guide the language model's attention. However, these approaches share a critical assumption: the dependency structure is predetermined and static throughout generation. This static view is limiting, as it fails to capture context-dependent feature relationships where one feature's value can dynamically alter dependencies among others. Our work challenges this assumption by proposing an adaptive framework where dependency structures evolve dynamically during synthesis.

3 Methodology

3.1 Problem Formulation

Let $\mathcal{T}$ be a tabular dataset with N samples, $t_i \in \mathcal{T}$ is represented by F features $(f_1, \ldots, f_F)$. Each feature f_j in sample t_i takes a specific value v_{ij}. Following the standard taxonomy in tabular toolbox (Patki et al., 2016), we categorize the features into two disjoint sets: $\mathcal{F}_{\text{num}}$ and $\mathcal{F}_{\text{cat}}$, such that the total set of features is $\mathcal{F} = \mathcal{F}_{\text{num}} \cup \mathcal{F}_{\text{cat}}$, where:

- $\mathcal{F}_{\text{num}}$: Continuous numerical variables, e.g. age.
- $\mathcal{F}_{\text{cat}}$: Discrete character variables with a finite set of possible values, e.g. marital status.

Our objective is to learn the underlying distribution of the samples in $\mathcal{T}$ and to generate a new set of M synthetic samples, denoted as $\hat{\mathcal{T}} = \{\hat{t}_1, \ldots, \hat{t}_M\}$, where $\hat{t} \notin \mathcal{T}$.

3.2 Tabular Data Generator

3.2.1 Embedding Tabular Records to Text.

To enable LLMs to process tabular data, we transform each sample t_i into a natural language sentence s_i. Specifically, we convert structured records into sequences of *<subject, predicate, object>* phrases, typically in the form of "*feature* is *value*" templates (Borisov et al., 2023). Therefore, for each feature f_j and its corresponding value v_{ij} in sample t_i, we convert the pair into a short phrase of the form "f_j is v_{ij}". All such phrases for a given sample are concatenated using commas to form the sentence s_i. By applying the template, we obtain a set of textual representations $S = \{s_1, \ldots, s_N\}$ corresponding to the original tabular dataset $\mathcal{T}$. We then fine-tune the LLM on S to model the underlying data distribution.

3.2.2 Training.

Following the continued pretraining strategy adopted in GREAT (Borisov et al., 2023), we fine-tune the LLM on S by minimizing the negative log-likelihood of each target token of values $(v_{i1}, ..., v_{ij})$ in s_i, as shown in Eq. (1).

$$\mathcal{L}_{\text{LM}}(\theta) = -\sum_{i=1}^{N} \sum_{t=1}^{|s_i^v|} \log P_\theta \left(s_{i,t}^v \mid s_{i,<t}^v \right), \quad (1)$$

where $s_{i,t}^v$ denotes the t-th value-related token in sentence s_i, and θ represents the model parameters.

To enhance the robustness of θ in modeling the feature-value distribution, we incorporate the permutation strategy from GRADE (Zhang et al., 2025), where the order of "*feature is value*" phrases in each s_i is randomly shuffled during training. This discourages the model from learning spurious dependencies that arise solely from fixed sequence positions, which do not reflect true feature dependencies in $\mathcal{T}$. After that, the fine-tuned generator θ implicitly integrates the knowledge encoded in the pretrained LLM with the feature-value distribution learned from $\mathcal{T}$. This enables it to model inter-feature dependencies and reduces the risk of logical inconsistencies in generated samples.

Consistent with GREAT and GRADE, θ generates synthetic samples by randomly selecting a subset of real feature-value pairs as a prefix, and then autoregressively completing the remaining ones. Formally, let $x_{1:k}$ denote the prefix consisting of k tokens derived from a subset of feature-value pairs. The model then generates the remaining

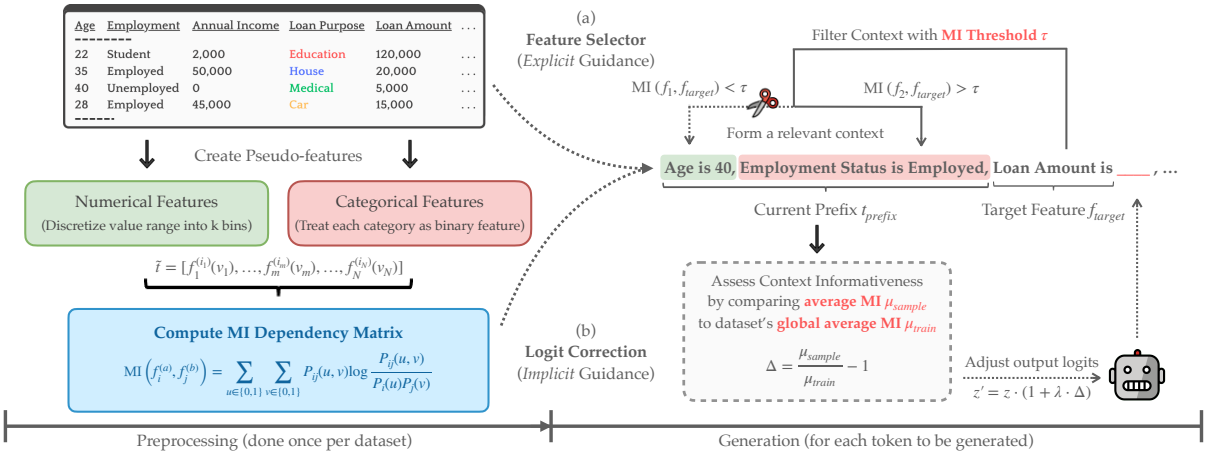

Figure 2: Overview of SAGE. In the preprocessing stage (**left**), a mutual-information-based dependency matrix is constructed from the data. During generation (**right**), this matrix guides the model using one of two strategies: (a) *Feature Selector*, which provides explicit guidance by pruning the input context with an MI threshold τ; and (b) *Logit Correction*, which provides implicit guidance by adaptively adjusting the output logits according to the informativeness of the current context.

tokens $x_{k+1:L}$ by sampling from the conditional distribution:

$$P_\theta(x_{k+1:L} \mid x_{1:k}) = \prod_{t=k+1}^{L} P_\theta(x_t \mid x_{1:t-1}), \quad (2)$$

where x_t denotes the t-th token in the generated sequence, and L is the total length of the output.

However, the attention mechanism forces each token to consider all prior tokens in s_i, even those unrelated to the current generation. As a result, θ can learn misleading dependencies between dependent features, leading to spurious correlations that degrade downstream model performance.

3.3 Generation guided by Sparse and Dynamic Feature Dependency

The core challenge lies in balancing sparsity and adaptability in dependency modeling. While existing methods either ignore feature relationships entirely or assume static connections, SAGE introduces a dynamic approach that adapts to feature values during generation.

Figure 2 illustrates the complete pipeline of our approach. SAGE operates in two phases: *preprocessing* and *generation*. During preprocessing, we discretize features into value-aware pseudo-features and construct a mutual information dependency matrix that captures statistical relationships between feature values. During generation, this matrix guides the LLM through two complementary strategies: (a) *Feature Selector* provides explicit guidance by filtering the input context with a mutual information (MI) threshold (Cover and Thomas, 2006), and (b) *Logit Correction* offers implicit guidance by adaptively adjusting output logits based on the context's informativeness. This dynamic approach enables the model to focus on truly relevant dependencies while adapting to value-conditioned patterns during synthesis.

Methods leveraging predefined feature graphs (Yang et al., 2025; Zhang et al., 2025) have demonstrated general superiority of modeling sparse dependencies over fully connected attention. However, we argue that these approaches fail to account for the dynamic nature of fine-grained feature-value dependencies during generation.

Formally, let $\mathcal{G} = (\mathcal{F}, \mathcal{E})$ represent a static feature dependency graph, where $\mathcal{F}$ denotes the set of features and $\mathcal{E} \subseteq \mathcal{F} \times \mathcal{F}$ encodes pairwise dependencies. However, when the value of a particular feature $f \in \mathcal{F}$ changes during generation, it may dynamically influence how other features relate to f, thereby modifying their relevance or conditional influence. As a result, this dynamic nature of feature interactions poses a challenge for approaches that rely on externally defined and static dependency graphs, as they struggle to adapt to evolving dependencies that emerge throughout the generative process. Consequently, static methods may suffer from reduced flexibility and accuracy, particularly in scenarios where the semantics of a given feature value significantly reshape the dependencies among remaining features.

As a solution, SAGE jointly accounts for the **sparsity** of feature dependencies and the **value-aware dynamics** of such dependencies during gen-

eration. To achieve this, we first discretize each feature f into k *pseudo-features* based on its value domain, denoted as:

$$\text{Bin}(f) = \{f^{(1)}, f^{(2)}, \ldots, f^{(k)}\}, \tag{3}$$

where each $f^{(i)}$ represents a bin corresponding to a specific sub-range or category of values for f.

Let $\Delta_f = \max(f) - \min(f)$ and $w_f = \Delta_f / k$. Set the cut-points $a_i = \min(f) + i\, w_f$ for $0 \le i \le k$. The k binary bins are

$$\begin{aligned} f^{(i)}(v) &= \mathbb{I}[\, a_{i-1} \le v < a_i \,], \quad (1 \le i < k), \\ f^{(k)}(v) &= \mathbb{I}[\, a_{k-1} \le v \le a_k \,]. \end{aligned} \tag{4}$$

Half-open intervals keep bins disjoint, and the right-closed last bin captures $v = \max(f)$. The number of bins k is set by the Freedman-Diaconis rule $k = \lceil \Delta_f / (2\,\text{IQR}(f)\, n^{-1/3}) \rceil$, capped at $k \le 16$ to control sparsity.

For categorical features $f \in \mathcal{F}_{\text{cat}}$, we treat each possible category $c \in \mathcal{V}_f$ as a separate pseudo-feature:

$$f^{(c)}(v) = \mathbb{I}[v = c], \tag{5}$$

where $\mathcal{V}_f$ is the set of all possible values of feature f. As a result, the values of a record $[v_1, \ldots, v_F]$ consisting of both numerical and categorical values is transformed, after binning, into an expanded binary vector of pseudo-features:

$$\tilde{t} = [f_1^{(i_1)}(v_1), \ldots, f_m^{(i_m)}(v_m), \ldots, f_F^{(i_F)}(v_F)], \tag{6}$$

where each $f_m^{(i_m)}(v_m)$ indicates whether the value v_m of feature f_m activates the corresponding bin or category. This binarized representation enables the model to capture fine-grained, value-dependent interactions between features while maintaining sparsity.

Finally, we compute the mutual information between pseudo-features to quantify their statistical dependency. Given two pseudo-features $f_i^{(a)}$ and $f_j^{(b)}$, their mutual information is defined as:

$$\sum_{u \in \{0,1\}} \sum_{v \in \{0,1\}} P_{ij}(u, v) \log \frac{P_{ij}(u, v)}{P_i(u) P_j(v)}, \tag{7}$$

where $P_{ij}(u, v)$ denotes the joint probability of activation values u and v for $f_i^{(a)}$ and $f_j^{(b)}$ respectively, while $P_i(u)$ and $P_j(v)$ denote the marginal probabilities. In practice, we estimate these probabilities empirically on the training split after pseudo-feature expansion. This makes the MI computation depend on binary pseudo-feature activations rather than raw numeric scales, while the capped bin count prevents excessively sparse or imbalanced pseudo-features from dominating the dependency graph. Appendix E further shows that the downstream performance is stable across a broad range of MI thresholds, indicating that the guidance mechanism is not brittle to moderate estimation noise.

Based on the computed dependencies between pseudo-features, we can infer whether specific value ranges of the original features in dataset $\mathcal{T}$ exhibit statistical correlations. These correlations are then used to construct a dynamic feature dependency graph, which serves as a guiding structure to control the sampling process of θ.

3.3.1 Feature Selector.

During sampling, we propose an explicit optimization strategy that filters out irrelevant feature-value pairs based on a static mutual information threshold. Specifically, for each target feature f_{target}, we define the set of relevant pseudo-features as:

$$\mathcal{R}(f_{\text{target}}) = \{\, f_j^{(b)} \in \tilde{t}_{\text{prefix}} \;\mid\; \text{MI}\big(f_j^{(b)}, f_{\text{target}}\big) > \tau \,\}. \tag{8}$$

where $\tilde{t}_{\text{prefix}}$ is the set of currently activated pseudo-features, and τ is a global threshold fixed before generation. By default we use a tuning-free setting and set τ to the median of MI values computed on the training set. This provides a robust scale for sparsification; the selected context remains instance- and step-dependent through $\mathcal{R}(f_{\text{target}})$.

Then, the generator θ conditions only on this relevant subset when generating the value for f_{target}:

$$P_\theta(v_{\text{target}} \mid \mathcal{R}(f_{\text{target}})). \tag{9}$$

This mechanism allows θ to dynamically adjust its attention over previously generated features for each generation step, thereby enabling fine-grained and value-sensitive dependency modeling.

3.3.2 Logit Correction.

While the selector module explicitly prunes irrelevant feature-value pairs, it also introduces a potential risk: the mutual information threshold τ is empirically set, and an overly strict threshold may suppress meaningful dependencies. As an alternative that avoids this issue, we propose an implicit logit correction mechanism.

Concretely, we compute the average mutual information between the current target feature f_{target}

and all previously generated pseudo-features in $\tilde{t}_{\text{prefix}}$:

$$\mu_{\text{sample}} = \frac{1}{|\tilde{t}_{\text{prefix}}|} \sum_{f_j^{(b)} \in \tilde{t}_{\text{prefix}}} \text{MI}(f_j^{(b)}, f_{\text{target}}). \quad (10)$$

We then compare this value with the dataset-wide average mutual information:

$$\Delta = \frac{\mu_{\text{sample}}}{\mu_{\text{train}}} - 1, \quad (11)$$

where μ_{train} denotes the expected mutual information between pseudo-features and f_{target} across the training corpus.

We correct the final generation logit z' for f_{target}:

$$z' = z \cdot (1 + \lambda \cdot \Delta). \quad (12)$$

Here z is the unnormalised logit for the candidate value of f_{target}; all other logits are masked. The value λ is a scaling hyperparameter. This correction sharpens the logit via softmax when the information provided by the prefix is highly relevant, i.e. $\Delta > 0$, and smooths it when the prefix contains little useful information, i.e. $\Delta < 0$ (Goodfellow et al., 2016). Therefore, the model adaptively adjusts its generation confidence based on the information content of the already generated context.

4 Experiment

4.1 Datasets

Binary Classification. The *Adult Income* dataset (Becker and Kohavi, 1996) comprises 16 demographic and occupational variables and is used to predict whether an individual's annual income exceeds a specified threshold. The *Home Equity Line of Credit* (HELOC) dataset (Oliabev, 2022) includes 24 credit-related attributes extracted from credit reports, aiming to predict whether individuals will fully repay their HELOC balance within two years. To simulate a high-dimensional feature setting, we employ the *Myocardial Infarction Complications* (MIC) dataset (Golovenkin et al., 2020), which involves predicting myocardial infarction complications from 110 biological features measured on the first and third days of hospitalization.
Multi-class Classification. The *Iris* dataset (Fisher, 1936) contains four numerical features measuring sepal and petal dimensions, aimed at classifying samples into iris species. The *CDC Diabetes Health Indicators* dataset (Burrows, 2017) is a large-scale clinical study comprising over 250,000 records with 35 features, including demographics and laboratory results, designed to classify patients as healthy, Type 1 diabetes, or Type 2 diabetes.
Regression. The *California Housing* dataset (Nugent, 2018) comprises 10 variables describing housing and geographic attributes. The task is to predict the median value of owner-occupied homes.

4.2 Evaluation Metrics

Downstream Utility. To evaluate the downstream utility of the synthetic datasets, we generated synthetic data with the same sample size as the original datasets. We then trained decision tree (DT) and random forest (RF) models for both classification and regression tasks, following the intended task types associated with each dataset. These models represent a diverse range of learning paradigms to ensure comprehensive evaluation. For classification tasks, we reported *accuracy* and *F1 score*, while for regression tasks, we reported *mean absolute percentage error (MAPE)*. The evaluation results are summarized in Table 1.
Data Fidelity. Following Xu et al. (2025), we assess data fidelity with *violation rates* under dataset-specific constraints. On California Housing, the violation rate is the probability that a generated data point lies outside the true geographical boundaries of California, providing a concrete measure of spatial constraint adherence. On Adult Income, we additionally evaluate a semantic consistency rule between *education* and *education-num*, where generated records that violate the canonical ordering of education levels and years of education are counted as invalid. This extends the evaluation beyond a single housing-boundary test. The Housing results are shown in Figure 4, and the multi-dataset constraint results are summarized in Appendix Table 3. Additionally, we visualize the spatial distribution of the synthetic samples based on their geographic coordinates, as illustrated in Figure 3.
Realism. To measure the realism of the synthetic data, we trained a support vector machine (SVM) classifier (Cortes and Vapnik, 1995) using 5-fold cross-validation to distinguish between the original dataset and the synthetic dataset. The classification accuracy of this model serves as a proxy for realism: lower accuracy indicates that the synthetic data more closely resembles the real data and is therefore harder to distinguish. The corresponding results are reported in Appendix Table 2.
Privacy Protection. In line with (Zhang et al.,

		Income		HELOC		Iris		Diabetes		MIC		Housing
		ACC ↑	F1 ↑	ACC ↑	F1 ↑	ACC ↑	F1 ↑	ACC ↑	F1 ↑	ACC ↑	F1 ↑	MAPE ↓
Original	DT	83.54	0.73	67.90	0.68	100	1.00	82.73	0.33	96.76	0.96	0.27
	RF	81.76	0.78	71.14	0.71	100	1.00	79.87	0.44	98.23	0.98	0.21
TVAE	DT	83.14	0.73	64.70	0.63	55.17	0.45	81.93	0.37	**96.76**	0.95	0.37
	RF	79.38	0.73	68.91	0.69	58.62	0.53	82.20	**0.42**	96.76	0.96	0.30
CTGAN	DT	78.27	0.60	63.14	0.63	10.34	0.09	**82.94**	0.30	96.47	0.95	0.71
	RF	80.74	0.73	37.93	0.29	41.38	**0.37**	82.18	0.36	96.76	0.95	0.50
TabSyn	DT	**83.32**	**0.75**	68.55	0.68	89.65	0.89	3.82	0.04	**96.76**	0.95	0.32
	RF	79.61	0.59	70.48	0.70	**100**	**1.00**	8.14	0.12	96.76	0.95	**0.23**
GReaT	DT	59.85	0.60	61.31	0.61	41.38	0.36	**82.94**	0.30	94.71	0.95	0.34
	RF	69.42	0.69	70.18	0.70	44.83	0.35	82.37	0.41	97.06	**0.96**	0.26
GraDe	DT	67.51	0.55	67.54	0.67	**96.55**	**0.97**	**82.94**	**0.37**	92.94	0.93	**0.31**
	RF	78.57	0.63	70.73	0.71	**100**	**1.00**	81.86	**0.42**	96.76	0.95	**0.23**
SPADA	DT	77.65	0.50	61.62	0.61	**96.55**	**0.97**	81.17	0.32	96.17	0.95	0.40
	RF	**81.62**	0.75	69.37	0.69	**100**	**1.00**	68.67	0.37	97.06	0.96	0.25
Ours (w/FS)	DT	80.66*	0.68*	68.45*	0.68*	**96.55***	**0.97***	**82.94**	0.30	96.47	0.95	0.34
	RF	78.58*	0.75*	**70.89**	**0.71**	**100***	**1.00***	80.15	0.41	97.05	0.95	0.25
Ours (w/LC)	DT	82.58*	0.72*	**69.52***	**0.69***	**96.55***	**0.97***	75.58	0.34	96.47	0.95	0.64
	RF	79.86*	**0.76***	70.58	0.71	**100***	**1.00***	80.72	0.40	**97.35**	**0.96**	0.40

Table 1: Performance of classifiers/regressors trained on synthetic data for downstream tasks. The best results are in **Bold**, and underline indicates the second-best. "Original" denotes models trained and tested on the real dataset $\mathcal{T}$, while all other methods train on synthetic data and test on real data. "ACC" denotes accuracy and "MAPE" denotes Mean Absolute Percentage Error. Values marked with an asterisk (*) are statistically significantly different from GReaT (paired t-test, $p < 0.05$).

2024), we evaluated privacy protection by computing the *Distance to Closest Record (DCR)* using the L1 norm (Boyd and Vandenberghe, 2004), measuring the proximity between synthetic samples and the nearest records in the original dataset. A *higher DCR* implies reduced resemblance to any real individual and thus stronger privacy guarantees. Conversely, a *lower DCR* reflects closer alignment with the real data distribution. The results are visualized in Appendix Figure 5.

4.3 Experimental Setup

In all experiments, we use a batch size of 8, the AdamW optimizer (Loshchilov and Hutter, 2019), and a learning rate of $1e{-}4$. For sampling, we adopt nucleus sampling (Holtzman et al., 2020) with $p = 0.95$, a temperature of 1.0, and set the maximum generation length equal to the maximum sequence length observed in the training set. The mutual information threshold is empirically set to the median mutual information value computed from the training data.

5 Result and Discussion

(1) Downstream Utility: Our method outperforms baselines, especially on small datasets. In Table 1, we show that SAGE consistently outperforms the baseline GReaT across almost all downstream tasks. The most significant improvement is seen on the `Adult` dataset, where our method achieves an increase of over 10 points in F1 score. Notably, GReaT performs poorly on smaller datasets such as `Iris`, where its accuracy reaches only 44.83%, in stark contrast to our 96.55%. We attribute this to the overfitting of GReaT, which only learns limited surface information of the training set and fails to capture meaningful patterns. In contrast, SAGE leverages mutual information to guide both the prefix construction and logits manipulation of the LLM, thereby shielding the model from being misled by superficial signals and resulting in improved robustness.

Compared to TabSyn, SAGE maintains more stable performance across datasets of varying sizes and dependency structures, demonstrating better generalization and robustness across diverse tabular domains. This stability is particularly evident in the consistent performance improvements across both classification and regression tasks.

(2) Data Fidelity: Both variants reduce violations, with stronger gains under different constraint types. We observe that SAGE achieves a notable reduction in violation rates compared to all baseline methods on California Housing, as shown in Figure 4. Among our two variants, *Logit*

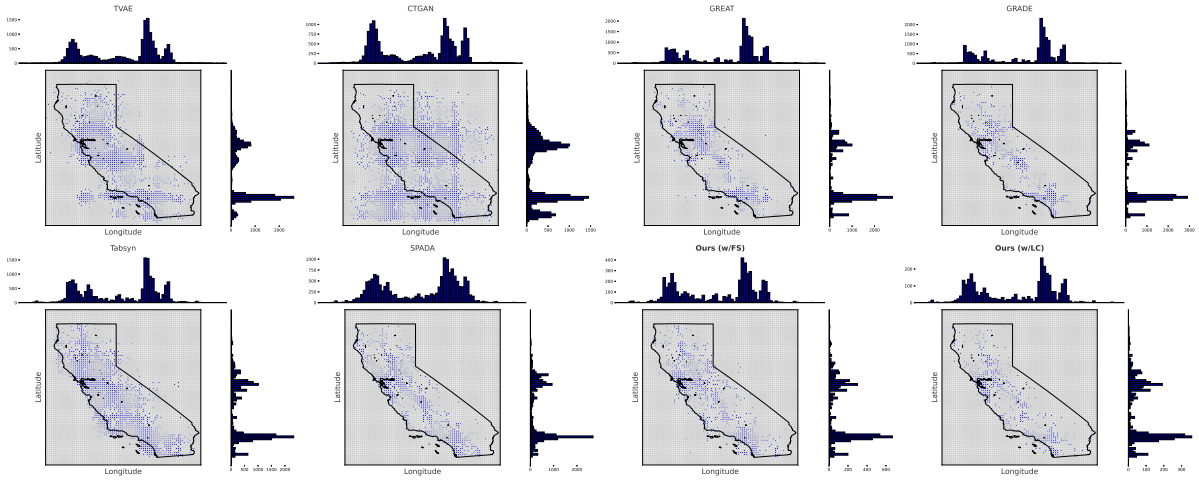

Figure 3: Comparison of the generated samples for the California Housing dataset, which includes characteristic information about various properties in California, USA. Joint histogram plots of the highly correlated variables Latitude and Longitude are shown. The black outline represents the true boundary of the state of California.

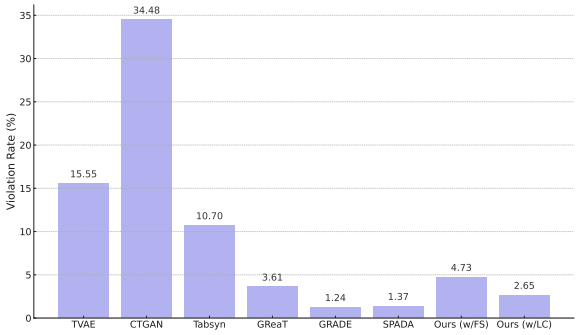

Figure 4: Violation rate, defined as the probability that a generated sample falls outside the true geographical boundaries of the state of California.

Correction achieves the largest improvement, reducing the violation rate by 1 point compared to GREAT, and achieving performance nearly on par with the GPT-4o-powered SPADA. Our *Feature Selector*-based variant reduces the violation rate by approximately 6% compared to TABSYN, demonstrating improved robustness and a superior ability to capture complex real-world data distributions.

As illustrated in the visualization in Figure 3, our method accurately captures the complex distribution of real-world data, with almost no synthetic samples falling outside the true boundaries of California. In contrast, TVAE and CTGAN struggle to learn such intricate spatial distribution patterns from limited training data and consequently fail to reproduce the correct spatial contours.

The additional semantic constraint evaluation on Adult Income further refines this picture. As shown in Appendix Table 3, *Feature Selector* reduces the education-consistency violation rate to 1.32%, substantially below most baselines and much lower than *Logit Correction*. This suggests that explicit context pruning is especially helpful when the target rule depends on a small set of semantically precise attributes, whereas implicit correction is more advantageous in the smoother spatial setting of Housing.

Takeaways. Compared to GREAT and SPADA, SAGE exhibits superior generalization on small-scale datasets such as *Iris*, suggesting that mutual information-based guidance effectively mitigates overfitting to superficial token-level patterns—a limitation of autoregressive LLM generators. Dynamic adaptation of dependency structures allows SAGE to focus on relevant feature relationships rather than spurious correlations.

When comparing the two proposed guidance strategies, we observe distinct advantages for each approach. *Logit Correction* achieves lower violation rates on Housing, indicating enhanced fidelity in spatially constrained domains where coherent relationships are crucial. *Feature Selector* generally yields lower MAPE in regression tasks and more stable classification performance, likely due to its explicit filtering of spurious contextual signals. This explicit filtering is particularly beneficial in high-dimensional feature spaces where noise significantly impacts generation quality.

These results demonstrate that explicit and implicit guidance mechanisms provide complementary benefits depending on domain characteristics. *Feature Selector* suits scenarios requiring aggressive noise filtering, while *Logit Correction* better handles complex interdependencies. We note that on HELOC, *Logit Correction* occasionally sup-

presses informative signals when contextual mutual information is underestimated, resulting in overly cautious generation. Addressing this through adaptive thresholding or hybrid strategies combining both approaches represents a promising direction for future work.

6 Conclusion

We introduce SAGE, a sparse and adaptive guidance framework for LLM-based tabular data generation that explicitly models the dynamic and sparse nature of feature dependencies. By discretizing features into pseudo-features and filtering context through mutual information, SAGE enables fine-grained and semantically accurate generation. Our method supports both explicit feature selection and implicit logit correction, offering flexible value-aware guidance during synthesis. Extensive experiments across six diverse datasets demonstrate that SAGE consistently improves generation quality, achieving up to +10.3% F1 improvement over the LLM-based baseline GREAT on Adult and reducing policy violation rates by over 6% on Housing. These results validate that value-sensitive dependency modeling leads to more realistic, controllable, and privacy-preserving synthetic tabular data.

Limitations

While SAGE demonstrates significant improvements in tabular data generation, we acknowledge several avenues for future work:

(1) Modeling of Higher-Order Dependencies. Our guidance mechanism relies on pairwise mutual information to construct the dependency graph, which may not explicitly capture more complex, higher-order interactions where multiple features collectively influence a target. However, SAGE's autoregressive generation process partially mitigates this limitation. By conditioning each new value on the entire sequence of previously generated feature-value pairs, the underlying LLM can implicitly learn and leverage these multi-feature contexts during synthesis, moving beyond the purely pairwise signals used for guidance.

(2) Scalability of Preprocessing. For datasets with extremely high dimensionality, the one-time preprocessing step of computing the mutual information matrix could become computationally intensive. Nevertheless, this design choice was made deliberately to maximize efficiency during the critical generation phase. Since this computation is performed only once per dataset, the subsequent synthesis process is highly scalable. The sparse context provided by our *Feature Selector* ensures that the inference cost remains low, avoiding the quadratic complexity of dense attention models at generation time.

Acknowledgments

This work was partially supported by the Verband der Vereine Creditreform e.V..

Use of AI Assistants The authors acknowledge the use of ChatGPT exclusively to refine the text in the final manuscript.

References

Barry Becker and Ronny Kohavi. 1996. Adult. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5XW20.

Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. 2022. Deep neural networks and tabular data: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–21.

Vadim Borisov, Kathrin Sessler, Tobias Leemann, Martin Pawelczyk, and Gjergji Kasneci. 2023. Language models are realistic tabular data generators. In *The Eleventh International Conference on Learning Representations*.

Stephen P Boyd and Lieven Vandenberghe. 2004. *Convex optimization*. Cambridge university press.

Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake Vanderplas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API design for machine learning software: experiences from the scikit-learn project. In *European Conference on Machine Learning and Principles and Practices of Knowledge Discovery in Databases*, Prague, Czech Republic.

Nilka Rios Burrows. 2017. Incidence of end-stage renal disease attributed to diabetes among persons with diagnosed diabetes—united states and puerto rico, 2000–2014. *MMWR. Morbidity and mortality weekly report*, 66.

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20:273–297.

Thomas M. Cover and Joy A. Thomas. 2006. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, USA.

Liancheng Fang, Aiwei Liu, Hengrui Zhang, Henry Peng Zou, Weizhi Zhang, and Philip S. Yu. 2025. Tabgen-icl: Residual-aware in-context example selection for tabular data generation. *CoRR*, abs/2502.16414.

Ronald A Fisher. 1936. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188.

S.E. Golovenkin, V.A. Shulman, D.A. Rossiev, P.A. Shesternya, S.Yu. Nikulina, Yu.V. Orlova, and V.F. Voino-Yasenetsky. 2020. Myocardial infarction complications. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C53P5M.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. 2016. *Deep learning*, volume 1. MIT press Cambridge.

Guangzeng Han, Weisi Liu, and Xiaolei Huang. 2025. Attributes as textual genes: Leveraging LLMs as genetic algorithm simulators for conditional synthetic data generation. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 19367–19389, Suzhou, China. Association for Computational Linguistics.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. 2025. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Sanket Kamthe, Samuel A. Assefa, and Marc Peter Deisenroth. 2021. Copula flows for synthetic data generation. *ArXiv*, abs/2101.00598.

Jayoung Kim, Jinsung Jeon, Jaehoon Lee, Jihyeon Hyeong, and Noseong Park. 2021. Oct-gan: Neural ode-based conditional tabular gans. In *Proceedings of the Web Conference 2021*, pages 1506–1515.

Jinhee Kim, Taesung Kim, and Jaegul Choo. 2024. Epic: Effective prompting for imbalanced-class data synthesis in tabular data classification via large language models. In *Advances in Neural Information Processing Systems*, volume 37, pages 31504–31542. Curran Associates, Inc.

Akim Kotelnikov, Dmitry Baranchuk, Ivan Rubachev, and Artem Babenko. 2023. TabDDPM: Modelling tabular data with diffusion models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 17564–17579. PMLR.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9).

Tennison Liu, Zhaozhi Qian, Jeroen Berrevoets, and Mihaela van der Schaar. 2023b. GOGGLE: Generative modelling for tabular data by learning relational structure. In *The Eleventh International Conference on Learning Representations*.

Tongyu Liu, Ju Fan, Guoliang Li, Nan Tang, and Xiaoyong Du. 2024. Tabular data synthesis with generative adversarial networks: design space and optimizations. *The VLDB Journal*, 33(2):255–280.

Yunbo Long, Liming Xu, and Alexandra Brintrup. 2025. Llm-tabflow: Synthetic tabular data generation with inter-column logical relationship preservation. *arXiv preprint arXiv:2503.02161*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *Preprint*, arXiv:1711.05101.

Dang Nguyen, Sunil Gupta, Kien Do, Thin Nguyen, and Svetha Venkatesh. 2024. Generating realistic tabular data with large language models. *CoRR*, abs/2410.21717.

Cameron Nugent. 2018. California housing prices. https://www.kaggle.com/datasets/camnugent/california-housing-prices.

Averkiy Oliabev. 2022. Home equity line of credit (heloc) dataset. https://www.kaggle.com/datasets/averkiyoliabev/home-equity-line-of-creditheloc.

Neha Patki, Roy Wedge, and Kalyan Veeramachaneni. 2016. The synthetic data vault. In *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 399–410.

Yubin Qu, Fang Li, Long Li, Xianzhen Dou, and Hongmei Wang. 2022. Can we predict student performance based on tabular and textual data? *IEEE Access*, 10:86008–86019.

Timur Sattarov, Marco Schreyer, and Damian Borth. 2023. Findiff: Diffusion models for financial tabular data generation. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, ICAIF '23, page 64–72, New York, NY, USA. Association for Computing Machinery.

Nabeel Seedat, Nicolas Huynh, Boris van Breugel, and Mihaela van der Schaar. 2024. Curated LLM: Synergy of LLMs and data curation for tabular augmentation in ultra low-data regimes.

Ruxue Shi, Yili Wang, Mengnan Du, Xu Shen, and Xin Wang. 2025. A comprehensive survey of synthetic tabular data generation. *CoRR*, abs/2504.16506.

Mihaela C. Stoian and Eleonora Giunchiglia. 2025. Beyond the convexity assumption: Realistic tabular data generation under quantifier-free real linear constraints. In *The Thirteenth International Conference on Learning Representations*.

Mihaela CÄ Stoian, Eleonora Giunchiglia, and Thomas Lukasiewicz. 2025. A survey on tabular data generation: Utility, alignment, fidelity, privacy, and beyond. *arXiv preprint arXiv:2503.05954*.

Namjoon Suh, Xiaofeng Lin, Din-Yin Hsieh, Merhdad Honarkhah, and Guang Cheng. 2023. Autodiff: combining auto-encoder and diffusion model for tabular data synthesizing. *CoRR*, abs/2310.15479.

Vibeke Binz Vallevik, Aleksandar Babic, Serena E. Marshall, Severin Elvatun, Helga M.B. Brøgger, Sharmini Alagaratnam, Bjørn Edwin, Narasimha R. Veeraragavan, Anne Kjersti Befring, and Jan F. Nygård. 2024. Can i trust my fake data – a comprehensive quality assessment framework for synthetic tabular data in healthcare. *International Journal of Medical Informatics*, 185:105413.

Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. 2019. Modeling tabular data using conditional gan. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Shengzhe Xu, Cho-Ting Lee, Mandar Sharma, Raquib Bin Yousuf, Nikhil Muralidhar, and Naren Ramakrishnan. 2024. Are llms naturally good at synthetic tabular data generation? *CoRR*, abs/2406.14541.

Shengzhe Xu, Cho-Ting Lee, Mandar Sharma, Raquib Bin Yousuf, Nikhil Muralidhar, and Naren Ramakrishnan. 2025. Why llms are bad at synthetic table generation (and what to do about it). *Preprint*, arXiv:2406.14541.

Shuo Yang, Chenchen Yuan, Yao Rong, Felix Steinbauer, and Gjergji Kasneci. 2024. P-TA: Using proximal policy optimization to enhance tabular data augmentation via large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 248–264, Bangkok, Thailand. Association for Computational Linguistics.

Shuo Yang, Zheyu Zhang, Bardh Prenkaj, and Gjergji Kasneci. 2025. Doubling your data in minutes: Ultra-fast tabular data generation via LLM-induced dependency graphs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 10348–10369, Suzhou, China. Association for Computational Linguistics.

Hengrui Zhang, Jiani Zhang, Zhengyuan Shen, Balasubramaniam Srinivasan, Xiao Qin, Christos Faloutsos, Huzefa Rangwala, and George Karypis. 2024. Mixed-type tabular data synthesis with score-based diffusion in latent space. In *The Twelfth International Conference on Learning Representations*.

Yishuo Zhang, Nayyar A. Zaidi, Jiahui Zhou, and Gang Li. 2021. Ganblr: A tabular data generation model. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 181–190.

Zheyu Zhang, Han Yang, Bolei Ma, David Rügamer, and Ercong Nie. 2023. Baby's CoThought: Leveraging large language models for enhanced reasoning in compact models. In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 158–170, Singapore. Association for Computational Linguistics.

Zheyu Zhang, Shuo Yang, Bardh Prenkaj, and Gjergji Kasneci. 2025. Not all features deserve attention: Graph-guided dependency learning for tabular data generation with language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 6217–6242, Suzhou, China. Association for Computational Linguistics.

Zilong Zhao, Robert Birke, and Lydia Y Chen. 2025. Tabula: Harnessing language models for tabular data synthesis. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 247–259. Springer.

Zilong Zhao, Aditya Kunar, Robert Birke, and Lydia Y. Chen. 2021. Ctab-gan: Effective table data synthesizing. In *Proceedings of The 13th Asian Conference on Machine Learning*, volume 157 of *Proceedings of Machine Learning Research*, pages 97–112. PMLR.

Zilong Zhao, Aditya Kunar, Robert Birke, Hiek Van der Scheer, and Lydia Y. Chen. 2024. CTAB-GAN+: enhancing tabular data synthesis. *Frontiers Big Data*, 6.

A Appendix

The structure of Appendix is as follows:

- Section B provides theoretical foundations for SAGE, including information-theoretic justifications for the *Feature Selector* and *Logit Correction* mechanisms.
- Section C presents engineering optimizations that reduce computational overhead and improve generation reliability through supervised fine-tuning and rejection sampling strategies.
- Section D reports additional experimental results on distributional fidelity, realism and privacy preservation metrics, including density distribution analysis, discriminator accuracy and distance to closest record evaluation.
- Section E conducts an ablation study examining the impact of mutual information thresholds on downstream performance across different datasets.
- Section F details implementation specifics, including evaluation settings and comprehensive dataset statistics.

B Theoretical Foundations of SAGE

In this section, we provide theoretical insights to motivate the design of our method SAGE, grounded in information theory and probabilistic modeling.

B.1 Mutual Information as a Proxy for Dependency

Let X and Y denote two (pseudo-)features from the transformed binary tabular space. The mutual information (MI) between X and Y is defined as:

$$I(X;Y) = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} P(x,y)\log\frac{P(x,y)}{P(x)P(y)}. \tag{13}$$

This quantity measures the reduction in uncertainty of one variable given knowledge of the other. In our context, a high MI between a context feature and the target feature indicates a strong statistical dependency. Hence, selecting features with high $I(X;Y)$ for conditioning naturally enhances the relevance and coherence of the generated value.

B.2 Feature Selector: An Information Bottleneck View

The *Feature Selector* aims to construct a reduced context $\mathcal{C} \subset \mathcal{F}_{\text{prefix}}$ such that:

$$\mathcal{C} = \arg\max_{\mathcal{C}\subset\mathcal{F}_{\text{prefix}}} \sum_{f_i\in\mathcal{C}} MI(f_i; f_{\text{target}}), \quad \text{s.t. } |\mathcal{C}| \le K, \tag{14}$$

where K is implicitly controlled by the mutual information threshold τ. This formulation resembles the *information bottleneck principle*, where one seeks to retain only the most informative subset of input variables for predicting the output while compressing irrelevant ones.

B.3 Logit Correction: KL Divergence Justification

Let $P^*(v \mid \mathcal{C})$ denote the optimal target value distribution given an ideal context $\mathcal{C}$, and let $P_\theta(v \mid \mathcal{C})$ be the model's predicted distribution. Minimizing the Kullback-Leibler divergence

$$D_{\text{KL}}(P^* \,||\, P_\theta) = \sum_v P^*(v \mid \mathcal{C}) \log\frac{P^*(v \mid \mathcal{C})}{P_\theta(v \mid \mathcal{C})} \tag{15}$$

is equivalent to maximizing the log-likelihood under P_θ. When P_θ underestimates the confidence due to weak context, we apply a logit rescaling based on mutual information strength:

$$z'_v = z_v \cdot (1+\lambda\cdot\Delta), \quad \text{with } \Delta = \frac{\mu_{\text{sample}}}{\mu_{\text{train}}} - 1, \tag{16}$$

where z_v is the pre-softmax logit of value v. This can be interpreted as dynamically adjusting P_θ to better approximate P^* when the context informativeness μ_{sample} deviates from expectation.

B.4 Dynamic Dependencies and Conditional Relevance

Finally, our binarization into value-aware pseudo-features enables approximation of conditional dependencies. For instance, mutual information between $f_i^{(a)}$ and $f_j^{(b)}$ (specific bins) reveals fine-grained value-conditioned patterns:

$$I(f_i^{(a)}; f_j^{(b)}) \approx \mathbb{E}_{v_i,v_j\sim T}\left[\log\frac{P(v_i,v_j)}{P(v_i)P(v_j)}\right]. \tag{17}$$

This allows SAGE to dynamically adapt generation based on the evolving feature prefix, approximating conditional distributions without requiring explicit Bayesian graphs or rule sets.

It is worth noting that the dynamic nature of SAGE lies in its flexible selection of dependent features; the use of fixed hyperparameters does not affect the inherent dynamism of the method itself.

Conclusion. These theoretical foundations justify the design of both the Feature Selector and Logit Correction components of SAGE, offering principled mechanisms for sparse and adaptive control of the generation process.

C Additional Engineering Optimization

C.1 Reduce Computational Overload

In the original GREAT framework (Borisov et al., 2023), LLMs are trained to generate not only the feature values but also the template text (e.g., "`feature is`"). We argue that this imposes unnecessary burden on the model and introduces redundant loss computations on tokens that are not informative for the target distribution.

To address this, we adopt a supervised fine-tuning (SFT) strategy that only computes loss over the tokens corresponding to feature values. During training, we mask out the template tokens and optimize the model solely on value tokens.

In inference, to remain consistent with the training objective, we select a random feature f and prepend its corresponding template phrase, i.e.,

Dataset	TVAE	CTGAN	GReaT	TabSyn	GRADE	SPADA	**Ours (w/FS)**	**Ours (w/LC)**
Income	80.23±0.01	74.08±0.02	99.95±0.00	**54.75±0.02**	99.98±0.00	65.83±0.03	73.80±0.01	88.28±0.01
HELOC	92.63±0.62	94.90±0.70	72.88±1.54	**54.15±2.49**	70.25±1.08	77.80±1.95	60.50±2.49	95.55±1.27
Iris	87.00±5.55	92.00±3.40	83.50±8.39	53.00±4.04	57.00±9.28	67.50±5.81	54.50±8.89	**51.50±3.45**
Diabetes	84.58±1.31	75.52±1.38	66.83±1.58	99.81±0.08	**64.82±0.63**	99.70±0.26	66.00±2.86	77.35±2.63
Housing	74.30±0.33	89.48±0.92	66.22±2.08	**50.28±0.95**	66.51±1.20	69.33±2.34	55.20±1.06	94.12±1.18
Mean (↓)	83.35	85.20	77.88	**53.55**	71.31	76.83	61.60	81.36

Table 2: Discriminator measure with a 5-fold cross-validation. Lower accuracy values indicate that the discriminator struggles to distinguish synthetic records from real data. **Bold** indicates the best performance, and underline indicates the second-best.

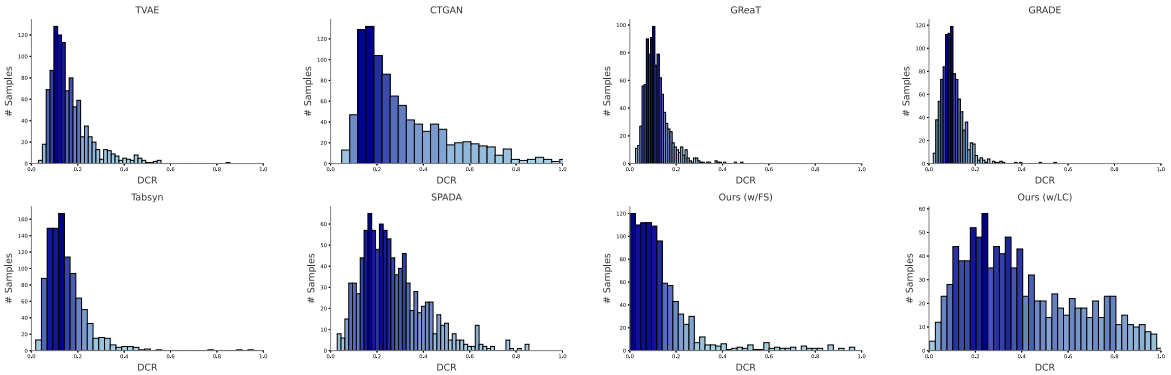

Figure 5: DCR for the California Housing dataset, evaluated with respect to the original training set. A lower DCR value demonstrates a high similarity between the synthetic data and the original data distribution, whereas a higher DCR value indicates enhanced privacy preservation in the synthetic data.

"`feature is`" as the prefix. The fine-tuned LLM is trained to fill in the missing value:

$$\hat{v}_f = \arg\max_{v_1,\ldots,v_T} \prod_{t=1}^{T} P_\theta(v_t \mid v_{<t}, \text{“}f \text{ is”}). \quad (18)$$

This design reduces the effective token-level computation by approximately $\sim 75\%$, as the model bypasses learning fixed template components. As a result, SAGE achieves significantly faster training and inference compared to GReaT, while maintaining comparable or better generation quality (see Section 5 for empirical results).

C.2 Rejection Sampling

Due to the inherent degeneration in text generation, LLM-based tabular data generators inevitably produce illegal values occasionally. For instance, a model may generate non-numeric strings for numeric variables, or generate out-of-domain tokens for categorical variables, i.e., tokens that do not appear in the training data for a given feature.

The GReaT framework applies heuristic rules to remove any generated feature values that contain invalid tokens. However, this approach not only leads to a loss of information from the original sample, but also introduces null-value noise into the dataset, which may negatively affect downstream learning tasks.

As a more efficient alternative, we adopt a rejection sampling strategy by explicitly constraining the output token space during decoding. Specifically, we restrict the model's vocabulary such that only tokens within the predefined valid range for each feature type can be sampled.

D Additional Experimental Results

Realism. To further demonstrate the effectiveness of SAGE, in Table 2, we report the discriminator measure with a 5-fold cross-validation. We also visualize the Distance to Closest Record (DCR) distributions for the California Housing dataset with respect to the original training set in Figure 5.

Constraint Fidelity Beyond Housing. Table 3 extends the violation-rate evaluation to two different kinds of constraints: a spatial boundary rule on California Housing and a semantic consistency rule on Adult Income. The results show that *Feature Selector* transfers particularly well to the semantic rule, whereas *Logit Correction* remains strongest on the spatial constraint, reinforcing that the two

guidance strategies are complementary.

Distributional Fidelity. To further validate the distributional quality of our synthetic data, we examine the density distributions of all four numerical features in the Iris dataset. Figure 6 compares the original data with outputs from both SAGE variants. The results demonstrate that *Feature Selector* and *Logit Correction* effectively capture the underlying patterns, with synthetic distributions closely matching the characteristic shapes and ranges of sepal and petal measurements. This visual evidence complements our quantitative metrics and confirms that SAGE maintains high fidelity across different feature types and scales.

Time Cost. Table 4 summarizes the training durations and average per-sample sampling times for both the baseline methods and our proposed models.

Performance Across Different LLMs. To demonstrate the robustness of SAGE across different model architectures, we evaluate both variants using GPT-2, Qwen-3, and Llama-3 as base models. Figure 7 presents results on Adult Income and California Housing datasets. Across all tested architectures, SAGE maintains consistent performance patterns, with Llama-3 generally achieving the best results.

Notably, the relative advantages of *Feature Selector* and *Logit Correction* remain stable across different models, indicating that our method's effectiveness is architecture-agnostic rather than dependent on specific LLM characteristics.

E Ablation Study: Impact of MI Thresholds

We examine how the MI threshold τ affects downstream performance by testing different threshold values on Adult Income and California Housing datasets. Figure 8 shows distinct patterns across the two datasets.

On Adult Income, performance remains relatively stable across most threshold ranges, with accuracy and F1 scores varying within 5 points. This stability suggests the dataset contains many genuinely irrelevant feature dependencies that can be safely filtered. California Housing shows different behavior. Performance stays stable until around 60-70% threshold, then degrades sharply. MAPE jumps from $\sim 25\%$ to over 50%, indicating that aggressive pruning removes important dependencies in this spatially-constrained domain.

These results reflect the datasets' inherent characteristics. Adult Income contains demographic features with clear independence, making sparse modeling effective. Housing data involves interconnected geographic and economic variables that require more careful dependency preservation. We set τ to the median MI value from training data (typically around 50%) as a reasonable balance between sparsity and information retention.

F Implementation Details

F.1 Evaluation Settings

For the machine learning efficiency and discriminator experiments, we additionally use decision tree (DT), random forest (RF), linear/logistic regression (LR) and support vector machine (SVM) models from the Scikit-Learn package (Buitinck et al., 2013).

F.2 Datasets

For all datasets, we use an 80%/20% train-test split for model training and evaluation. Table 5 provides comprehensive statistics for these datasets.

F.3 Hyperparameter

To ensure the reproducibility of our reported experimental results, we list the hyperparameters used for each dataset in Tab. 6. Furthermore, we employed nucleus sampling with $p = 0.7$ and temperature $= 1$ during decoding.

Dataset	TVAE	CTGAN	GREAT	TABSYN	SPADA	**Ours (w/FS)**	**Ours (w/LC)**
Income	4.21±0.64	34.41±1.18	**0.00±0.00**	2.32±0.49	3.59±0.98	1.32±1.16	16.47±4.51
Housing	15.55±0.55	34.48±0.72	3.61±0.28	10.70±0.72	5.26±0.62	4.73±0.59	**2.65±0.59**
Mean (↓)	9.88	34.40	**1.81**	6.51	4.41	3.03	9.56

Table 3: Violation rates under two dataset-specific constraints. For Adult Income, we measure the consistency between *education* and *education-num*; for California Housing, we measure whether generated coordinates fall outside the California boundary. Lower is better.

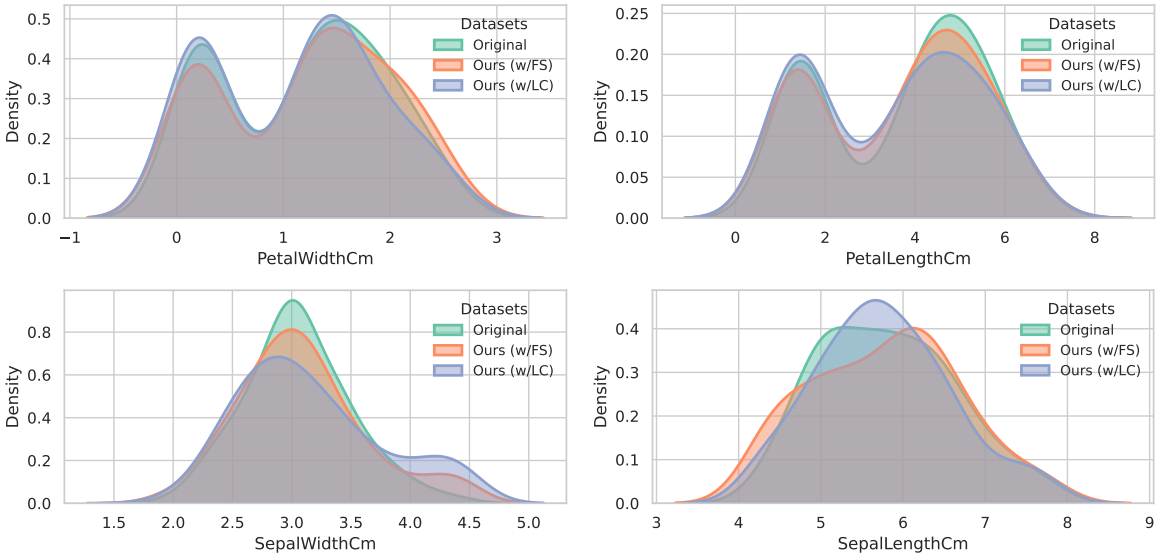

Figure 6: Visualization of the density distributions of Sepal and Petal lengths and widths on the Iris dataset, comparing the original and synthetic data.

Dataset		GREAT	**Ours (w/FS)**	**Ours (w/LC)**
Income	Training	6 h 10 min	15 min	15 min
	Sampling	9 sec	0.2 sec	0.2 sec
HELOC	Training	1 h 47 min	1 h 12 min	1 h 12 min
	Sampling	45 sec	0.5 sec	0.5 sec
Iris	Training	17 sec	20 sec	20 sec
	Sampling	4 sec	0.07 sec	0.07 sec
Housing	Training	1 h 18 min	52 min	52 min
	Sampling	8 sec	0.4 sec	0.4 sec

Table 4: Average end-to-end training time, including one-time preprocessing for pseudo-feature construction and MI estimation, and sampling time per instance.

Dataset	Domain	# Samples	# Features	Task	# Classes
Income (Becker and Kohavi, 1996)	Social	48,842	15	Classification	2
HELOC (Oliabev, 2022)	Finance	10,459	24	Classification	2
Iris (Fisher, 1936)	Biology	150	5	Classification	3
Diabetes (Burrows, 2017)	Healthcare	253,680	20	Classification	3
MIC (Golovenkin et al., 2020)	Biology	1,360	111	Classification	2
Housing (Nugent, 2018)	Real Estate	20,640	10	Regression	-

Table 5: The statistics of the datasets employed in our experiments. # Samples, # Features and # Classes denote the numbers of samples, features and classes in tabular datasets, respectively.

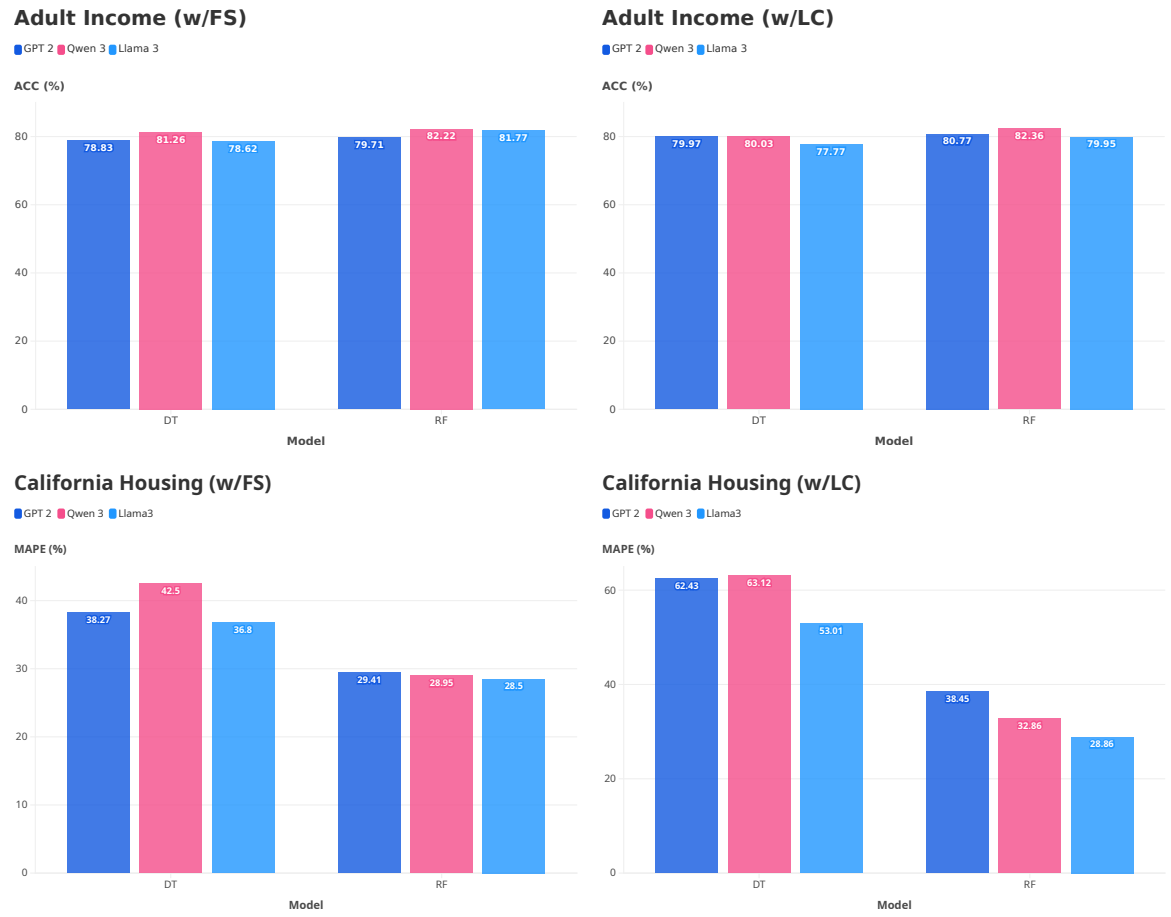

Figure 7: The performance of SAGE with different LLMs on classification and regression tasks.

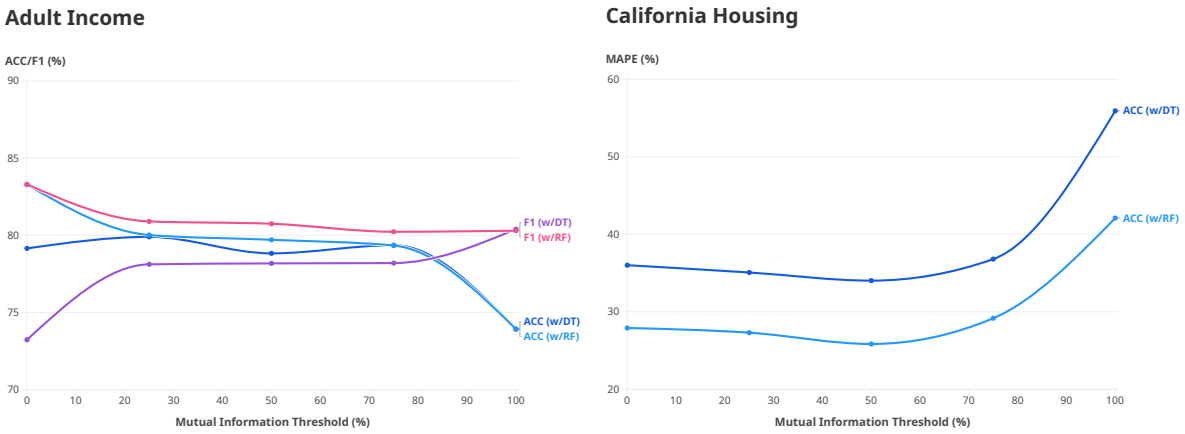

Figure 8: Impact of MI thresholds on downstream performance. The MI threshold refers to the proportion of training samples with the lowest MI; for instance, a 25% threshold considers the bottom 25% of samples sorted in ascending order by mutual information.

	Income	HELOC	Iris	Diabetes	MIC	Housing
Bins	5	10	5	20	20	10
MI threshold	0.0004	0.0142	0.1190	0.0063	0.0004	0.004
λ	1.0	1.0	1.2	0.8	0.8	1.0

Table 6: The hyperparameters used in our experiments. We performed a grid search over λ in the range of 0.8 to 1.2 with a step size of 0.2, and over the number of bins in the range of 5 to 20 with a step size of 5.