Not All Features Deserve Attention: Graph-Guided Dependency Learning for Tabular Data Generation with Language Models

Zheyu Zhang Shuo Yang Bardh Prenkaj Gjergji Kasneci

Technical University of Munich

{name.surname}@tum.de

Abstract

Large Language Models (LLMs) have shown strong potential for tabular data generation by modeling textualized feature-value pairs. However, tabular data inherently exhibits sparse feature-level dependencies, where many feature interactions are structurally insignificant. This creates a fundamental mismatch as LLMs' self-attention mechanism inevitably distributes focus across all pairs, diluting attention on critical relationships, particularly in datasets with complex dependencies or semantically ambiguous features. To address this limitation, we propose **GraDe** (**Gra**ph-Guided **De**pendency Learning), a novel method that explicitly integrates sparse dependency graphs into LLMs' attention mechanism. GraDe employs a lightweight dynamic graph learning module guided by externally extracted functional dependencies, prioritizing key feature interactions while suppressing irrelevant ones. Our experiments across diverse real-world datasets demonstrate that GraDe outperforms existing LLM-based approaches by up to 12% on complex datasets while achieving competitive results with state-of-the-art approaches in synthetic data quality. Our method is minimally intrusive yet effective, offering a practical solution for structure-aware tabular data modeling with LLMs. [1]

1 Introduction

Tabular data forms the foundations of countless real-world applications across healthcare analytics (Fatima et al., 2017), financial modeling (Dastile et al., 2020), demographic research (Ding et al., 2021), and scientific experimentation (Shwartz-Ziv and Armon, 2022). Generating high-quality synthetic tabular data has become increasingly important for data augmentation, privacy preservation, and model testing (Van Breugel and Van Der Schaar, 2024). However, this task presents unique challenges due to the complex structural properties inherent to tabular datasets.

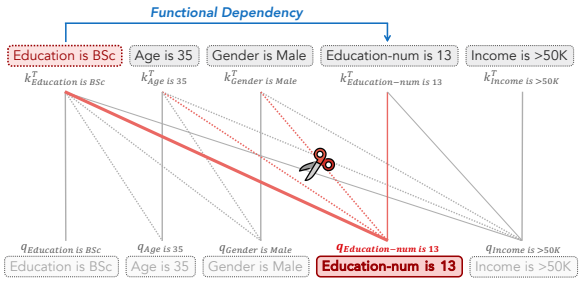

Figure 1: Structural mismatch between LLMs' dense attention patterns and tabular data's sparse dependency structure.

Unlike images or text, tabular data exhibits distinct patterns of relationships between features. Some features have strong correlations, while others remain entirely independent from one another (Wang et al., 2024). More critically, certain features may uniquely determine the values of others through logical or business rules. Database theory formalizes these patterns as *Functional Dependencies (FDs)*, where knowing the value of one feature (or a set of features) allows us to determine another's with certainty (Liu et al., 2012). For example, postal codes uniquely determine cities and states, while having no connection to personal attributes like age or income. Without preserving these dependencies, synthetic data suffers from logical inconsistencies such as customer IDs linked to multiple conflicting personal details, rendering the generated data unrealistic for practical applications.

Recent years have seen rapid progress in tabular data generation (Xu et al., 2019; Liu et al., 2023b; Kotelnikov et al., 2023; Zhang et al., 2024). Among these, Large Language Models (LLMs) offer a compelling solution due to their strong generalization capabilities from pre-training on vast corpora and understanding of complex real-world relationships (Liu et al., 2023a; Zhang et al., 2023b). Borisov et al. (2023) first applied LLMs to this task by con-

[1] Our code is available at https://github.com/oooranz/GraDe.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 6217–6242
November 4-9, 2025

verting data into text sequences (e.g., "*Age is 39, Income is* $\leq$*50K*") and randomly shuffling feature orders during training, allowing the model to generate data conditioned on any subset of features. Xu et al. (2024) later showed that this random shuffling disrupts the model's ability to capture dependencies between features, proposing a fixed topological ordering approach instead. This reveals a fundamental design challenge: how to maintain both flexible conditioning and accurate modeling of feature relationships in LLM-based approaches.

The root cause of this challenge is a *structural mismatch* between LLMs and tabular data. As shown in Figure 1, LLMs use attention mechanisms where every token potentially relates to every other token (Vaswani et al., 2017). In contrast, tabular data has sparse, non-sequential dependencies where most features are conditionally independent. This fundamental disconnect creates a dilemma: random feature ordering offers flexibility but weakens dependency modeling, while fixed ordering better captures relationships but sacrifices flexibility. When feature-value pairs are linearized as text, LLMs face the inherently difficult task of correctly interpreting each pair while preventing interference from unrelated features (Yan et al., 2024). To fully leverage LLMs for tabular data, we need an approach that maintains flexible ordering while explicitly modeling the sparse dependency structure inherent in tabular data.

To bridge this structural gap, we propose **GraDe** (**Gra**ph-Guided **De**pendency Learning). Unlike previous approaches that rely on implicit learning through feature ordering, GraDe explicitly enhances LLMs by incorporating learnable sparse dependency graphs into the attention mechanism (§4.2). Our method dynamically models token-level relationships while leveraging externally extracted functional dependencies as guidance (§4.3), allowing focused attention on structurally important connections. The model is trained with a multi-objective loss balancing language fluency, structural sparsity, and functional dependency alignment (§4.4). For parameter efficiency, we also introduce **GraDe-Light**, a variant that only updates our attention modules while maintaining competitive performance. Overall, our **contributions** are as follows:

- We formalize the structural mismatch problem between LLMs' dense attention patterns and tabular data's sparse dependency structure, providing a framework for understanding current methodological limitations (§3).
- We develop a graph-guided attention mechanism that dynamically models sparse dependencies between tokens and integrates functional dependency knowledge as soft supervision, enhancing structural awareness while preserving the underlying LLM architecture (§4).
- We empirically demonstrate that both GraDe and GraDe-Light show significant improvements over existing LLM-based approaches while achieving competitive results with state-of-the-art methods. Our approach performs particularly well on datasets with complex dependency structures and maintains advantages in low-resource settings (§6).

2 Related Work

Deep Generative Models for Tabular Modeling Early research on tabular data generation adapted deep generative models such as GANs, VAEs, and diffusion models (Shwartz-Ziv and Armon, 2022). CTGAN (Xu et al., 2019) and CTAB-GAN (Zhao et al., 2021) tackled mixed-type data through conditional sampling and specialized normalization strategies (Grinsztajn et al., 2022), while TVAE (Xu et al., 2019) employed a variational framework for more stable training. More recent diffusion-based methods like TabDDPM (Kotelnikov et al., 2023) and TabSyn (Zhang et al., 2024) improved sample quality through iterative denoising of multi-modal distributions. Notably, GOGGLE (Liu et al., 2023b) introduced explicit graph-based modeling of feature dependencies. Despite these advances, most generative approaches still struggle to flexibly capture the sparse dependency structures inherent in tabular data.

Language Models for Tabular Modeling Another line of research models tabular data by serializing rows into textual sequences and applying large language models for generation. GReaT (Borisov et al., 2023) pioneered this direction by serializing feature-value pairs as text and employing random feature permutations to enhance flexibility. However, this randomization weakens the model's ability to learn structural dependencies (Müller et al., 2024). Later works such as TabLLM (Hegselmann et al., 2023) and TAPTAP (Zhang et al., 2023a) proposed refined serialization formats and large-scale tabular pretraining to better preserve relational patterns. Recent work such as

P-TA (Yang et al., 2024) further explored enhancing LLM-based tabular generation through policy optimization, while SPADA (Yang et al., 2025) uses LLMs to build sparse feature dependencies in tabular data. These efforts highlight the need for stronger structural guidance beyond text serialization alone.

Injecting Structural Information into LMs The importance of aligning model architecture with data structure has been increasingly recognized across domains (Kitaev et al., 2020; Tay et al., 2021; Bi et al., 2025b,a). Various approaches address this alignment: sparse attention models (Child et al., 2019; Beltagy et al., 2020) constrain token interactions to relevant neighborhoods; graph-enhanced methods (Yang et al., 2021) incorporate explicit relational structures; and specialized architectures like GraSAME (Yuan and Färber, 2024) inject structural information directly into attention mechanisms. These developments reflect growing evidence that standard dense attention inadequately captures the inherent sparsity in real-world data (Li et al., 2022). Our GraDe method extends this insight to tabular modeling by introducing a graph-guided attention mechanism that leverages functional dependencies to focus on meaningful feature relationships while preserving the generative flexibility of language models.

3 Problem Formulation

Modeling Tabular Data as Language Generative modeling for tabular data aims to learn a probability distribution p_X over tabular samples $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ (Borisov et al., 2024). Given a training dataset $\mathcal{D}$ consisting of N i.i.d. samples $\mathbf{x} \sim p_X$, the objective is to approximate p_X by learning parameters θ of a generative distribution p_θ. While traditional approaches model this distribution directly in the feature space, language model-based methods typically transform tabular data into textual representations to model the joint distribution as:

$$p_\theta(\mathbf{x}) = \prod_{t=1}^{T} p_\theta(w_t \mid w_{<t}), \tag{1}$$

where $w_{1:T} = w(\mathbf{x})$ denotes a linearized textual sequence of feature-value pairs, such as *"Feature$_1$ is x_1, ..., Feature$_d$ is x_d"*.

Attention as Graph Structure In transformer-based language models, the self-attention mechanism can be treated as a fully connected graph over input tokens (Yao and Wan, 2020). For a sequence of tokens $\{t_1, t_2, ..., t_n\}$, the attention weight a_{ij} between tokens t_i and t_j is computed as:

$$a_{\text{ij}} = \text{softmax}\left(\frac{\mathbf{q}_i \mathbf{k}_j^\top}{\sqrt{d_k}}\right), \tag{2}$$

where $\mathbf{q}_i$ and $\mathbf{k}_j$ are the query and key vectors associated with tokens t_i and t_j. This mechanism can be interpreted as defining a fully connected directed graph $G_{\text{LM}} = (V, E)$, where $V = \{t_1, ..., t_n\}$ represents the tokens, and $E = \{(t_i, t_j) \mid i, j \in [1, n]\}$ denotes edges with weights a_{ij}. This structure assumes that any token can directly influence any other token, which is suitable for natural language where long-range dependencies are frequent and contextually relevant (Chaudhari et al., 2021).

Relational Structure of Tabular Data In contrast, tabular data typically exhibits sparse relational structures, where each feature depends only on a small subset of other features (Papenbrock et al., 2015). Such relationships can be represented as a sparse graph $G_{\text{Tab}} = (V', E')$, where vertices $V' = \{f_1, f_2, ..., f_d\}$ represent features, and edges $E' \subset V' \times V'$ represent pairwise dependencies, with $|E'| \ll |V'|^2$. Formally, the generative process of each feature value v_i can be described as:

$$v_i = \psi_i\left(N(i), \varepsilon_i\right), \tag{3}$$

where $N(i)$ is a small set of features that f_i depends on, ψ_i is a functional mapping, and ε_i denotes independent noise. As empirically observed by Liu et al. (2023b), such sparse relational structures are characteristic of real-world data generation processes.

Structure Mismatch Problem When applying language models to tabular data generation, a critical mismatch arises between the dense attention structure G_{LM} and the sparse relational structure G_{Tab}. This mismatch leads to **two key problems**: (1) The attention mechanism assigns non-negligible weights across all token pairs, diluting its ability to capture the few critical dependencies as the number of features increases; (2) Language models do not encode any inductive bias about tabular feature dependencies, and must learn such structural patterns purely from data, which becomes increasingly difficult as table complexity grows.

Motivated by these limitations, we propose to explicitly incorporate functional dependency structures into language models, as detailed in the following sections.

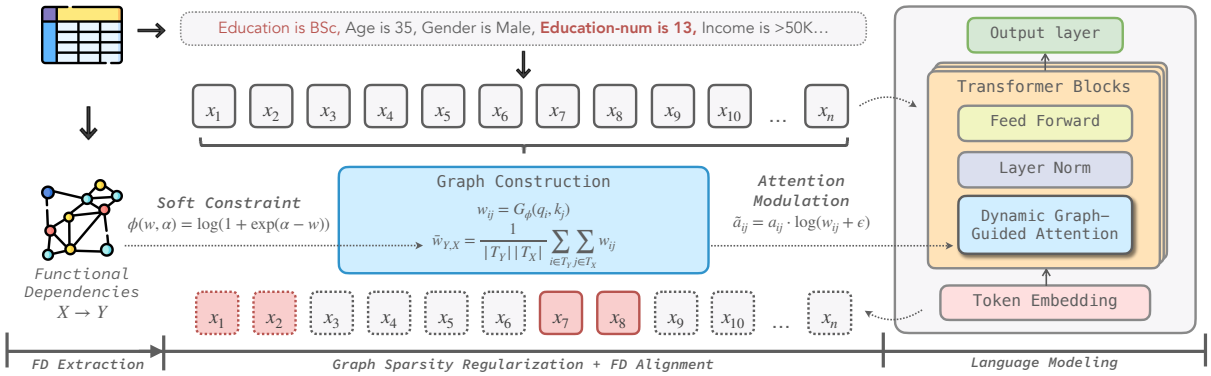

Figure 2: Pipeline of GraDe. Following Borisov et al. (2023), we first linearize tabular data into text with randomly permuted feature orders, and then feed the sequence into a language model. A dynamic token-level dependency graph is learned during training, guided by soft constraints from functional dependencies, and used to modulate attention weights for structure-aware tabular modeling.

4 Graph-Guided Dependency Learning

We propose GraDe, a novel method that enhances language models to better capture the sparse dependency structures inherent in tabular data. Building on the observation that tabular data contains natural sparsity in feature relationships, GraDe integrates this structural information into the attention mechanism of language models.

4.1 Overview

Recent work has established sparse dependencies in tabular data can be more accurately captured through relational structures (Liu et al., 2023b). While language models' self-attention mechanisms create relational representations among tokens, we hypothesize that their dense connectivity pattern, where every token attends to all others, may dilute focus on the critical relationships in tabular data.

GraDe addresses this challenge by learning a dynamic sparse graph that guides the attention mechanism to prioritize structurally important dependencies. Our approach incorporates three key innovations: (1) a graph-guided attention that dynamically models sparse token-level relationships; (2) integration of externally derived functional dependencies as supervision; and (3) sparsity regularization to focus attention on meaningful connections. Unlike previous methods, GraDe introduces minimal modifications to only the attention modules while preserving the semantic capabilities of pretrained language models. Figure 2 shows our architecture.

4.2 Dynamic Graph-Guided Attention

Graph Construction For a sequence of tokens, GraDe learns a weighted directed graph for each attention head, represented by an adjacency matrix $\mathbf{W} \in [0,1]^{n\times n}$, where each entry w_{ij} quantifies the importance of the directed relation from token j to token i. The graph is parameterized by a lightweight two-layer neural network G_ϕ:

$$w_{ij} = G_\phi(q_i, k_j) \tag{4}$$
$$= \sigma\left(W_2 \cdot \mathrm{ReLU}(W_1 \cdot [q_i; k_j])\right), \tag{5}$$

where $q_i, k_j \in \mathbb{R}^{d_k}$ are the query and key vectors, $W_1 \in \mathbb{R}^{d_k \times 2d_k}$ and $W_2 \in \mathbb{R}^{1\times d_k}$ are learnable parameters, and σ denotes the sigmoid activation. This module operates over the head dimension d_k, which leads to parameter efficiency.

Attention Modulation Standard self-attention computes alignment scores via query-key dot products (Brauwers and Frasincar, 2023), treating all token pairs as equally relevant candidates for attention. To encourage the model to focus on important relations, GraDe modulates these scores by incorporating the learned dependency weights through a logarithmic gating mechanism:

$$\tilde{a}_{ij} = a_{ij} \cdot \log(w_{ij} + \epsilon), \tag{6}$$

where a_{ij} is the unnormalized attention score, w_{ij} is the learned edge weight, and ϵ is a small constant for numerical stability. This transformation acts as a gating mechanism: when $w_{ij} \approx 1$ (indicating a strong dependency), it preserves the original score; when $w_{ij} \approx 0$, it strongly suppresses it. The modulated attention scores $\tilde{a}_{ij}$ are subsequently normalized via softmax, ensuring a valid attention distribution.

During inference, we optimize graph computation for the autoregressive setting by computing

only the edges involving the newly generated token, enabling efficient decoding for long sequences.

4.3 Incorporating Functional Dependencies

While the dynamic graph module captures token-level dependencies within the sequence, tabular data primarily exhibits important relationships at the feature level (Yan et al., 2024). These relationships often take the form of functional dependencies, where one set of features uniquely determines another.

To bridge this gap between token-level and feature-level dependencies, we incorporate prior knowledge about FDs into the training objective. For each known FD $X \to Y$, we compute the average connection strength from tokens representing features in X to those representing features in Y:

$$\bar{w}_{Y,X} = \frac{1}{|T_Y||T_X|} \sum_{i \in T_Y} \sum_{j \in T_X} w_{ij}, \tag{7}$$

where T_X and T_Y are the sets of token indices corresponding to features X and Y, respectively.

Based on these average connection strengths, we define an FD alignment loss:

$$\mathcal{L}_{\text{FD}} = \sum_{X \to Y \in \mathcal{F}} \phi(\bar{w}_{Y,X}, \alpha), \tag{8}$$

where $\mathcal{F}$ is the set of known FDs and ϕ is a constraint function enforcing a minimum strength threshold α.

We implement this constraint using a smooth softplus formulation:

$$\phi(w, \alpha) = \log(1 + \exp(\alpha - w)), \tag{9}$$

This formulation enables flexible modeling of functional dependencies while accommodating the inherent variability in real-world data. By smoothly penalizing insufficient connection strengths, the model is encouraged to focus on key structural relationships without imposing overly rigid constraints.

4.4 Multi-Objective Fine-Tuning

GraDe is trained using a composite loss that jointly optimizes for language modeling quality, structural sparsity, and alignment with known dependencies:

Language Modeling To preserve the sequential generation capability of language models, we adopt standard autoregressive training with the cross-entropy loss:

$$\mathcal{L}_{\text{LM}} = -\sum_{i=1}^{|x|} \log P(x_i \mid x_{<i}), \tag{10}$$

where x is the linearized input sequence derived from tabular data.

Graph Sparsity Regularization To reflect the inherent sparsity in tabular data relationships, we impose an L1 penalty on the learned adjacency matrices:

$$\mathcal{L}_{\text{sparse}} = \frac{1}{L} \sum_{l=1}^{L} \|\mathbf{W}^{(l)}\|_1, \tag{11}$$

where L is the number of transformer layers and $\mathbf{W}^{(l)}$ is the graph matrix at layer l. This regularization encourages the model to assign non-zero weights only to structurally meaningful dependencies, and avoid dense attention allocation.

FD Alignment The FD alignment loss (as given in Eq. 8) guides the model to respect known feature-level constraints by encouraging strong connections between dependent feature tokens.

Overall Training Loss The overall training loss combines the language modeling objective, sparsity regularization, and FD alignment:

$$\mathcal{L} = \mathcal{L}_{\text{LM}} + \lambda_{\text{sparse}} \cdot \mathcal{L}_{\text{sparse}} + \lambda_{\text{FD}} \cdot \mathcal{L}_{\text{FD}}, \tag{12}$$

where λ_{sparse} and λ_{FD} are hyperparameters that control the influence of sparsity and dependency alignment, respectively.

This multi-objective approach enables GraDe to learn representations that are both effective for language modeling and structurally aligned with the underlying tabular data dependencies.

Parameter-Efficient Variant We also introduce GraDe-Light, a lightweight variant that updates only the dynamic graph-guided attention modules while freezing all other parameters. This approach significantly reduces trainable parameters while maintaining most of the model's dependency modeling capability. We evaluate both variants to assess trade-offs between computational efficiency and generation quality.

5 Experimental Setup

Models We use GPT-2 (Radford et al., 2019) with 124 million parameters as our backbone model for all main experiments. We evaluate both the full GraDe and GraDe-Light variants on this model. Additional experiments with GPT-2 Medium with 355 million parameters are reported in Appendix A.3. We use nucleus sampling (Holtzman et al., 2020) for text generation.

Dataset		Original	TVAE	CTGAN	CTABGAN+	TabSyn	GReaT	**GraDe**	**GraDe-Light**
Bird (↑)	DT	99.97±0.02	70.59±0.00	94.28±0.17	66.46±0.21	99.75±0.06	99.63±0.07	99.70±0.00	99.78±0.02
	RF	100.00±0.00	79.87±0.46	99.09±0.08	75.54±0.11	99.97±0.12	100.00±0.00	100.00±0.00	100.00±0.00
	LR	100.00±0.00	87.04±0.00	100.00±0.00	73.99±0.00	100.00±0.00	98.29±0.00	100.00±0.00	98.85±0.00
Sick (↑)	DT	98.70±0.10	95.87±0.26	86.28±1.20	92.40±0.42	74.65±1.05	91.42±0.61	96.05±0.42	92.19±0.37
	RF	98.46±0.18	95.34±0.24	94.97±0.00	94.99±0.26	96.21±0.34	96.40±0.15	98.04±0.15	97.01±0.22
	LR	89.54±0.00	94.57±0.00	58.01±0.00	82.65±0.00	66.23±0.00	83.31±0.00	90.60±0.00	91.72±0.00
Income (↑)	DT	81.14±0.03	80.28±0.12	79.87±0.17	76.61±0.31	80.73±0.13	79.09±0.14	80.94±0.19	79.63±0.12
	RF	85.15±0.15	82.62±0.13	82.25±0.13	83.38±0.18	80.22±0.11	83.68±0.12	84.05±0.06	83.65±0.19
	LR	80.45±0.00	78.99±0.00	79.19±0.00	77.07±0.00	81.14±0.00	80.19±0.00	80.94±0.00	80.02±0.00
Diabetes (↑)	DT	74.68±1.30	72.21±0.49	59.48±0.88	62.99±1.23	75.06±2.08	65.97±0.66	78.05±0.66	67.08±1.40
	RF	74.94±0.88	75.97±0.71	57.01±1.50	64.81±1.50	75.32±0.92	67.40±0.76	77.27±1.42	71.56±1.76
	LR	69.48±0.00	73.38±0.00	57.14±0.00	72.08±0.00	70.78±0.00	67.53±0.00	68.83±0.00	66.88±0.00
Housing (↓)	DT	0.24±0.01	0.35±0.00	0.50±0.00	0.39±0.00	0.30±0.01	0.34±0.00	0.27±0.00	0.31±0.01
	RF	0.18±0.00	0.30±0.01	0.37±0.00	0.28±0.01	0.22±0.00	0.24±0.01	0.21±0.01	0.23±0.00
	LR	0.29±0.00	0.33±0.00	0.35±0.00	0.29±0.00	0.29±0.00	0.30±0.00	0.31±0.00	0.31±0.00

Table 1: Machine learning efficiency experiment. The best results are marked in **Red**. "Original" refers to training on original real data. Classification accuracy (↑) and regression mean absolute percentage error (↓) are reported.

Datasets For our main experiments, we use five real-world datasets from diverse domains with complex dependencies: **Bird** (Ecology), **Sick** and **Diabetes** (Medical), **Income** (Social), and **Housing** (Real Estate). Dataset sizes range from 700 to 40,000 samples with 5 to 30 features, including continuous, categorical, and integer types. For our scaling experiments (§A.1), we use the **Loan** dataset, a large-scale financial dataset with approximately 250,000 samples. Detailed descriptions and the dataset statistics are provided in Appendix B.3.

FDs Extraction Functional dependency detection is a mature, well-studied problem in the database domain (Liu et al., 2012). We use HyFD (Papenbrock and Naumann, 2016), a hybrid algorithm for functional dependency discovery, to automatically extract FDs from our datasets. We verify the results using TANE (Huhtala et al., 1999). The detected FDs serve as prior knowledge for tabular data modeling. Further details on the detection algorithms and the extracted FDs for each dataset are provided in Appendix B.4.

Baselines We compare GraDe with five representative tabular data generation methods: VAE-based TVAE (Xu et al., 2019), which employs mode-specific normalization; GAN-based methods CTGAN (Xu et al., 2019) and CTABGAN+ (Zhao et al., 2024), where CTGAN uses mode-specific normalization for handling complex distributions and CTABGAN+ introduces downstream losses and specialized encoders for mixed variables; diffusion-based method TabSyn (Zhang et al., 2024), which operates in a latent space to capture inter-column relationships; and LLM-based method GReaT (Borisov et al., 2023).

Evaluation Following prior works on tabular data generation (Borisov et al., 2023; Gulati and Roysdon, 2023; Liu et al., 2023b), we evaluate synthetic data quality across three key dimensions: (1) **Utility**: we assess whether synthetic data can effectively replace real data by training models on synthetic data and evaluating them on real test data (Esteban et al., 2017); (2) **Fidelity**: we measure how well the statistical dependencies between attributes are preserved, which is often more critical than matching individual distributions; and (3) **Privacy**: we verify that generated data resembles but doesn't duplicate training examples by computing distances between synthetic samples and their closest records in the original dataset. For all evaluations, we report average results over five runs with different random seeds to ensure reproducibility.

6 Results

6.1 Synthetic Data Quality

Machine Learning Efficiency (MLE) We evaluate the practical *utility* of synthetic data by training discriminative models on the generated datasets and testing their performance on real test set. This setup reflects whether the synthetic data can effectively replace real data in downstream learning tasks. To ensure generality across model classes, we use three commonly adopted classifiers: Decision Tree (DT), Random Forest (RF) (Ho, 1995), and Linear/Logistic Regression (LR). Table 1 shows MLE scores across datasets compared to models trained on the original training dataset. GraDe demonstrates strong performance, with par-

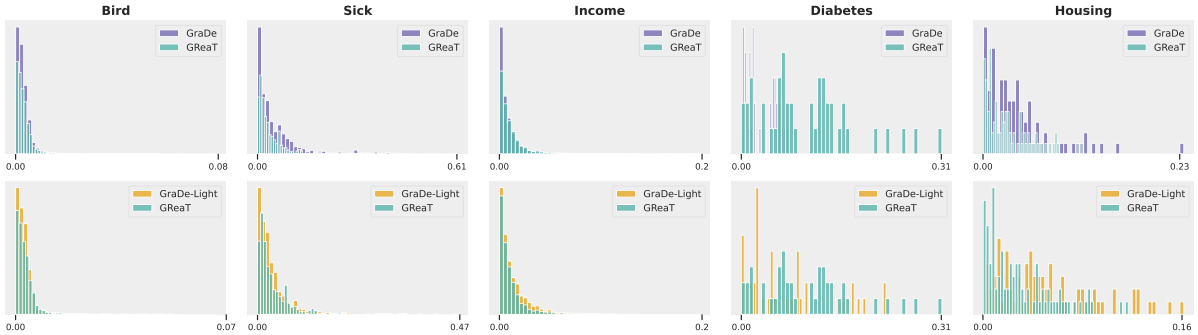

Figure 3: Correlation error histograms across five datasets. Each bar represents the absolute difference between real and synthetic data correlation coefficients, with values closer to zero indicating better fidelity. GraDe (purple) and GraDe-Light (yellow) show error distributions more concentrated near zero compared to GReaT (green).

Method	Bird	Sick	Income	Diabetes	Housing	Average
TVAE	.434±.032	.262±.023	.156±.025	.360±.010	.165±.006	.275±.019
CTGAN	.180±.022	.702±.040	.560±.044	.783±.020	.259±.010	.497±.027
CTABGAN+	.279±.027	.386±.032	.533±.036	.516±.014	.197±.008	.382±.023
TabSyn	.008±.004	.202±.019	.397±.033	.385±.014	.152±.006	.229±.015
GReaT	.010±.005	.098±.011	.165±.059	.368±.012	.117±.004	.152±.018
GraDe	.002±.001	.081±.009	.161±.019	.323±.012	.103±.003	.134±.009
GraDe-Light	.007±.004	.145±.065	.147±.018	.355±.014	.112±.003	.153±.021

Table 2: Distance to closest record (DCR) scores (lower is better). The best results are marked in **Red**.

ticularly significant improvements over GReaT (our closest LLM-based competitor) on complex medical datasets, achieving gains of up to 12% on Diabetes and 5% on Sick. These results highlight the advantage of explicit dependency modeling in domains with complex structural relationships. Our GraDe-Light variant achieves similar performance while using 100 million fewer trainable parameters than GReaT, highlighting the efficiency of our graph-guided approach.

Column-Wise Correlation Error The preservation of feature relationships is crucial for generating realistic tabular data. Following Gulati and Roysdon (2023), we assess *fidelity* by measuring the absolute difference between Pearson correlation coefficients in real versus synthetic data. For categorical features, we first apply one-hot encoding before calculating correlations. Figure 3 shows the correlation error histograms comparing GraDe, GraDe-Light, and GReaT across different datasets. We focus this comparison on LLM-based approaches since prior work has shown that autoregressive models typically struggle with capturing joint distributions (Zhang et al., 2024), making this a particularly challenging aspect for LLM-based tabular data generation. Our results demonstrate that GraDe's graph-guided approach provides a clear advantage by explicitly modeling functional dependencies and directing attention to structurally relevant connections.

Distance to the Closest Record (DCR) *Privacy* preservation is a key application for synthetic tabular data, requiring generated samples to resemble but not duplicate training examples. Following Zhang et al. (2024), we use the L1 norm to measure the minimum distance between each synthetic sample and real training examples. For numerical features, differences are normalized by the observed feature range; for categorical features, the distance is defined as 0 if the values match, and 1 otherwise. The final DCR score is computed as the average of the minimum distances across all synthetic samples. Table 2 reports the DCR scores across all five datasets. While most models perform adequately on simpler datasets, our approach shows substantial improvements on medical datasets (Sick, Diabetes) with complex dependencies. This enhancement is particularly valuable in sensitive domains, as lower DCR scores indicate greater dissimilarity from training records and are associated with reduced membership inference risks (Hu et al., 2022). These results demonstrate how explicit dependency modeling enables language models to better balance fidelity and privacy.

6.2 Modeling in Low-Data Regimes

Data-oriented generative approaches like GAN or diffusion-based methods typically struggle in data-scarce scenarios, requiring substantial training examples to capture complex distributions (Manousakas and Aydöre, 2023). In contrast, LLM-based methods can leverage rich pre-trained knowledge to model tabular data effectively even with limited samples (Seedat et al., 2024; Gardner et al., 2024). We evaluate this capability on Bird and Income datasets using training sets from 250 to 4000 examples, comparing GraDe against GReaT on identical data subsets. Figure 4 shows GraDe consistently outperforms GReaT across all data regimes, with gains up to 15% under extreme scarcity (250 examples). Although the gap narrows with more data, GraDe maintains its advantage throughout. GraDe-Light achieves comparable improvements despite fewer parameters. We observe more pronounced improvements on the Income dataset, where the extracted functional dependencies are both more numerous and structurally complex. This empirical pattern suggests that GraDe benefits more from structural guidance when the underlying dependencies are richer or harder to learn implicitly.

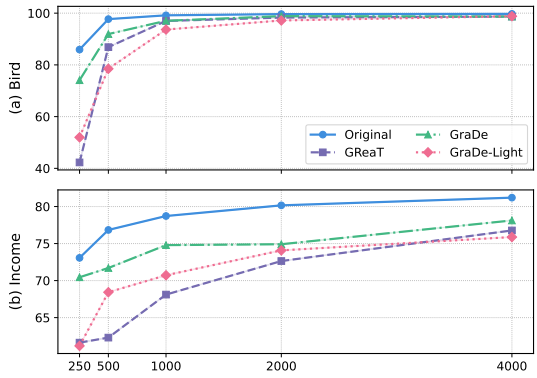

Figure 4: Classification accuracy across different training sample sizes in low-data regimes. "Original" denotes models trained on original real data; others are trained on synthetic data.

6.3 Ablation Study

We assess the contribution of each component in our approach through ablation experiments with two variants: (1) GraDe w/o Sparsity, which removes graph sparsity regularization, and (2) GraDe w/o FDs, which removes FD alignment loss. Table 3 compares these variants against the full GraDe model and GReaT across all datasets using decision trees trained on generated data.

	Bird Acc ↑	**Sick** Acc ↑	**Income** Acc ↑	**Diabetes** Acc ↑	**Housing** MAPE ↓
GraDe	99.70	96.05	80.94	78.05	0.27
w/o Sparsity	+0.23	−0.40	−0.51	−0.63	+0.01
w/o FDs	−0.03	−6.51	−0.51	−5.47	−0.03
GReaT	−0.07	−4.63	−1.85	−12.08	−0.07

Table 3: Ablation study across five datasets comparing GraDe, its ablated variants, and the GReaT baseline. Accuracy (Acc) is used for classification tasks and mean absolute percentage error (MAPE) for regression tasks.

Both components enhance GraDe's effectiveness, with FD alignment having the stronger impact. Removing FD alignment significantly degrades performance on medical datasets (Sick and Diabetes), where numerical attributes and ambiguous feature names make implicit learning challenging. For datasets with clearer dependencies (Bird, Income, Housing), the impact is smaller, and removing sparsity regularization occasionally improves performance, possibly because the attention distribution benefits from added flexibility in datasets with strong local patterns. These results demonstrate that explicit dependency guidance is most valuable for complex data structures, while both components work complementarily to optimize overall performance, providing empirical support for our hypothesis that dense attention may dilute focus on critical relationships.

7 Conclusion

In this work, we address the structural mismatch between LLMs' dense attention patterns and tabular data's sparse dependencies. Our solution, GraDe, guides token interactions through learnable dependency graphs that integrate functional relationships into attention mechanisms. This approach helps models focus on meaningful connections without altering the base architecture. Experiments across diverse datasets show consistent gains in utility, fidelity, and privacy. GraDe outperforms prior LLM-based methods by up to 12% on complex medical datasets and remains effective even with only 250 training samples. The lightweight GraDe-Light variant achieves similar results while reducing trainable parameters by nearly 100 million. These findings confirm that structural inductive bias significantly enhances LLMs for tabular modeling. By bridging the gap between powerful semantic representations and domain-specific structural awareness, GraDe offers a practical and effective solution for generating high-quality synthetic tabular data.

8 Limitations

While GraDe effectively enhances tabular data modeling through functional dependencies, several limitations exist. First, extremely high-dimensional tables may exceed LLM context limits and increase computational costs. While GraDe-Light reduces parameters, sequential generation of wide tables remains inefficient. Future models with larger contexts may naturally address these scaling issues.

Second, our method relies on externally extracted functional dependencies, which may inherit dataset biases or miss subtle relationships. However, automatic FD extraction provides scalable, domain-agnostic supervision without expert annotation. GraDe mitigates bias through soft constraints, though critical applications may benefit from expert verification.

Finally, training converges slower than baseline models because our graph-guided attention modules train from scratch while language model components use pretrained weights. The graph structures need additional iterations to align with pretrained representations. Future work could explore transfer learning or better initialization for faster convergence.

Acknowledgments

This work was partially supported by the Verband der Vereine Creditreform e.V.. We thank Andreas Kotowski and his colleagues from Creditreform for their valuable collaboration and for providing useful discussions and insights that contributed to this work.

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

References

Alfred V. Aho. 1990. Algorithms for finding patterns in strings. In Jan van Leeuwen, editor, *Handbook of Theoretical Computer Science, Volume A: Algorithms and Complexity*, pages 255–300. Elsevier and MIT Press.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *CoRR*, abs/2004.05150.

Jinhe Bi, Yifan Wang, Danqi Yan, Xun Xiao, Artur Hecker, Volker Tresp, and Yunpu Ma. 2025a. Prism: Self-pruning intrinsic selection method for training-free multimodal data selection. *arXiv preprint arXiv:2502.12119*.

Jinhe Bi, Yujun Wang, Haokun Chen, Xun Xiao, Artur Hecker, Volker Tresp, and Yunpu Ma. 2025b. LLaVA steering: Visual instruction tuning with 500x fewer parameters through modality linear representation-steering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15230–15250, Vienna, Austria. Association for Computational Linguistics.

Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. 2024. Deep neural networks and tabular data: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 35(6):7499–7519.

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*.

Gianni Brauwers and Flavius Frasincar. 2023. A general survey on attention mechanisms in deep learning. *IEEE Trans. on Knowl. and Data Eng.*, 35(4):3279–3298.

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.

Sneha Chaudhari, Varun Mithal, Gungor Polatkan, and Rohan Ramanath. 2021. An attentive survey of attention models. *ACM Trans. Intell. Syst. Technol.*, 12(5).

Jiawei Chen, Xinyan Guan, Qianhao Yuan, Guozhao Mo, Weixiang Zhou, Yaojie Lu, Hongyu Lin, Ben He, Le Sun, and Xianpei Han. 2025. Consistentchat: Building skeleton-guided consistent dialogues for large language models from scratch. *arXiv preprint arXiv:2506.03558*.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. *CoRR*, abs/1904.10509.

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

Xolani Dastile, Turgay Celik, and Moshe Potsane. 2020. Statistical and machine learning models in credit scoring: A systematic literature survey. *Applied Soft Computing*, 91:106263.

Frances Ding, Moritz Hardt, John Miller, and Ludwig Schmidt. 2021. Retiring adult: New datasets for fair machine learning. In *Advances in Neural Information Processing Systems*, volume 34, pages 6478–6490. Curran Associates, Inc.

Dheeru Dua and Casey Graff. 2017. UCI machine learning repository.

Cristóbal Esteban, Stephanie L. Hyland, and Gunnar Rätsch. 2017. Real-valued (medical) time series generation with recurrent conditional gans. *CoRR*, abs/1706.02633.

Dominique Evans-Bye. 2015. States shapefile. https://hub.arcgis.com/datasets/CMHS::states-shapefile/explore?location=29.721532%2C71.941464%2C3.76.

Meherwar Fatima, Maruf Pasha, et al. 2017. Survey of machine learning algorithms for disease diagnostic. *Journal of Intelligent Learning Systems and Applications*, 9(01):1.

Joshua P Gardner, Juan Carlos Perdomo, and Ludwig Schmidt. 2024. Large scale transfer learning for tabular data via language modeling. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Sergey E Golovenkin, Jonathan Bac, Alexander Chervov, Evgeny M Mirkes, Yuliya V Orlova, Emmanuel Barillot, Alexander N Gorban, and Andrei Zinovyev. 2020. Trajectories, bifurcations, and pseudo-time in large clinical datasets: applications to myocardial infarction and diabetes data. *GigaScience*, 9(11):giaa128.

Leo Grinsztajn, Edouard Oyallon, and Gael Varoquaux. 2022. Why do tree-based models still outperform deep learning on typical tabular data? In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Manbir S Gulati and Paul F Roysdon. 2023. TabMT: Generating tabular data with masked transformers. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Guangzeng Han, Weisi Liu, and Xiaolei Huang. 2025. Attributes as textual genes: Leveraging llms as genetic algorithm simulators for conditional synthetic data generation. *Preprint*, arXiv:2509.02040.

Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. 2023. Tabllm: Few-shot classification of tabular data with large language models. In *International conference on artificial intelligence and statistics*, pages 5549–5581. PMLR.

Tin Kam Ho. 1995. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE.

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*.

Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S. Yu, and Xuyun Zhang. 2022. Membership inference attacks on machine learning: A survey. *ACM Comput. Surv.*, 54(11s).

Yká Huhtala, Juha Kärkkäinen, Pasi Porkka, and Hannu Toivonen. 1999. Tane: An efficient algorithm for discovering functional and approximate dependencies. *The Computer Journal*, 42(2):100–111.

Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

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.

Kunchang Li, Yali Wang, Gao Peng, Guanglu Song, Yu Liu, Hongsheng Li, and Yu Qiao. 2022. Uniformer: Unified transformer for efficient spatial-temporal representation learning. In *International Conference on Learning Representations*.

Jixue Liu, Jiuyong Li, Chengfei Liu, and Yongfeng Chen. 2012. Discover dependencies from data—a review. *IEEE Transactions on Knowledge and Data Engineering*, 24(2):251–264.

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*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Xuan Lu, Sifan Liu, Bochao Yin, Yongqi Li, Xinghao Chen, Hui Su, Yaohui Jin, Wenjun Zeng, and Xiaoyu Shen. 2025. Multiconir: Towards multi-condition information retrieval. *Preprint*, arXiv:2503.08046.

James Lucas, George Tucker, Roger Grosse, and Mohammad Norouzi. 2019. Understanding posterior collapse in generative latent variable models.

Dionysis Manousakas and Sergül Aydöre. 2023. On the usefulness of synthetic tabular data generation. *CoRR*, abs/2306.15636.

Luis Müller, Mikhail Galkin, Christopher Morris, and Ladislav Rampásek. 2024. Attending to graph transformers. *Trans. Mach. Learn. Res.*, 2024.

R Kelley Pace and Ronald Barry. 1997. Sparse spatial autoregressions. *Statistics & Probability Letters*, 33(3):291–297.

Thorsten Papenbrock, Jens Ehrlich, Jannik Marten, Tommy Neubert, Jan-Peer Rudolph, Martin Schönberg, Jakob Zwiener, and Felix Naumann. 2015. Functional dependency discovery: an experimental evaluation of seven algorithms. *Proc. VLDB Endow.*, 8(10):1082–1093.

Thorsten Papenbrock and Felix Naumann. 2016. A hybrid approach to functional dependency discovery. In *Proceedings of the 2016 International Conference on Management of Data*, SIGMOD '16, page 821–833, New York, NY, USA. Association for Computing Machinery.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Lv Qingsong, Yangning Li, Zihua Lan, Zishan Xu, Jiwei Tang, Yinghui Li, Wenhao Jiang, Hai-Tao Zheng, and Philip S Yu. 2025. Raise: Reinforenced adaptive instruction selection for large language models. *arXiv preprint arXiv:2504.07282*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Jun Rao, Zepeng Lin, Xuebo Liu, Xiaopeng Ke, Lian Lian, Dong Jin, Shengjun Cheng, Jun Yu, and Min Zhang. 2025. Apt: Improving specialist llm performance with weakness case acquisition and iterative preference training. *arXiv preprint arXiv:2506.03483*.

Jun Rao, Xuebo Liu, Lian Lian, Shengjun Cheng, Yunjie Liao, and Min Zhang. 2024. CommonIT: Commonality-aware instruction tuning for large language models via data partitions. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10064–10083, Miami, Florida, USA. Association for Computational Linguistics.

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.

Ravid Shwartz-Ziv and Amitai Armon. 2022. Tabular data: Deep learning is not all you need. *Inf. Fusion*, 81(C):84–90.

Jack W Smith, James E Everhart, William C Dickson, William C Knowler, and Robert Scott Johannes. 1988. Using the adap learning algorithm to forecast the onset of diabetes mellitus. In *Proceedings of the annual symposium on computer application in medical care*, page 261.

Yi Tay, Dara Bahri, Donald Metzler, Da-Cheng Juan, Zhe Zhao, and Che Zheng. 2021. Synthesizer: Rethinking self-attention for transformer models.

Boris Van Breugel and Mihaela Van Der Schaar. 2024. Position: Why tabular foundation models should be a research priority. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 48976–48993. PMLR.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Alex X. Wang, Stefanka S. Chukova, Colin R. Simpson, and Binh P. Nguyen. 2024. Challenges and opportunities of generative models on tabular data. *Applied Soft Computing*, 166:112223.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

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. Why llms are bad at synthetic table generation (and what to do about it). *arXiv preprint arXiv:2406.14541*.

Jiahuan Yan, Bo Zheng, Hongxia Xu, Yiheng Zhu, Danny Chen, Jimeng Sun, Jian Wu, and Jintai Chen. 2024. Making pre-trained language models great on tabular prediction. In *The Twelfth International Conference on Learning Representations*.

Junhan Yang, Zheng Liu, Shitao Xiao, Chaozhuo Li, Defu Lian, Sanjay Agrawal, Amit S, Guangzhong Sun, and Xing Xie. 2021. Graphformers: GNN-nested transformers for representation learning on textual graph. In *Advances in Neural Information Processing Systems*.

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. *arXiv preprint arXiv:2507.19334*.

Shaowei Yao and Xiaojun Wan. 2020. Multimodal transformer for multimodal machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4346–4350, Online. Association for Computational Linguistics.

Shuzhou Yuan and Michael Färber. 2024. GraSAME: Injecting token-level structural information to pretrained language models via graph-guided self-attention mechanism. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 920–933, Mexico City, Mexico. 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*.

Tianping Zhang, Shaowen Wang, Shuicheng Yan, Li Jian, and Qian Liu. 2023a. Generative table pre-training empowers models for tabular prediction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14836–14854, Singapore. Association for Computational Linguistics.

Zheyu Zhang, Han Yang, Bolei Ma, David Rügamer, and Ercong Nie. 2023b. 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.

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 Additional Results

This section extends our evaluation with supplementary experiments, providing deeper insights into GraDe's effectiveness across diverse scenarios. Specifically, we evaluate the approach on a large-scale imbalanced dataset (§A.1), measure intrinsic constraint preservation in generated data (§A.2), and investigate scalability to a larger backbone model (§A.3). Additionally, we assess data fidelity using discriminator-based measures (§A.4), examine sensitivity to functional dependency alignment weights (§A.5), and evaluate the impact of sparsity constraint on high-dimensional data (§A.6).

A.1 Scaling to Large-Scale Imbalanced Data

Real-world large-scale datasets, particularly in financial domains such as loan default prediction, commonly exhibit severe imbalance issues, with minority classes often being critically important yet underrepresented. To evaluate GraDe's performance on such challenges, we conducted additional machine learning efficiency experiments on a large-scale loan dataset containing approximately 250,000 samples with only 12% positive cases predicting loan defaults based on customer behavior. This experiment directly assesses the utility of synthetic data generated by different methods when applied to large imbalanced datasets.

Table 4 presents AUC and F1 scores as evaluation metrics, which better represent performance on imbalanced data compared to accuracy. The results reveal distinct performance patterns across different methods. TVAE completely failed to generate minority class samples, likely due to posterior collapse (Lucas et al., 2019). While CTGAN and CTABGAN+ managed to capture some minority patterns, their performance remains limited with AUC scores barely above random chance. TabSyn demonstrates strong performance, particularly with RF and LR models, leveraging its latent space modeling approach. GraDe achieves the best performance with DT models and remains competitive across other classifiers, outperforming other LLM-based methods. These results highlight GraDe's effectiveness at capturing complex dependencies even in challenging imbalanced scenarios, where modeling minority classes accurately is crucial for downstream applications.

	DT		RF		LR	
	AUC ↑	F1 ↑	AUC ↑	F1 ↑	AUC ↑	F1 ↑
Original	.739±.000	.347±.000	.814±.002	.434±.003	.635±.000	.268±.000
TVAE	-	-	-	-	-	-
CTGAN	.505±.001	.211±.001	.522±.001	.210±.002	.521±.000	.210±.000
CTABGAN+	.521±.001	.217±.000	.538±.002	.222±.001	.543±.000	.224±.000
TabSyn	.571±.001	.246±.001	.645±.001	.269±.002	.594±.000	.244±.000
GReaT	.569±.000	.254±.000	.622±.002	.252±.002	.553±.000	.224±.000
GraDe	.587±.001	.271±.000	.642±.003	.267±.004	.581±.000	.238±.000
GraDe-Light	.576±.001	.260±.000	.611±.004	.245±.002	.550±.000	.218±.000

Table 4: Additional results for machine learning experiment on the large-scale imbalanced **Loan** dataset (with approx. 250K samples and 12% positive class). The best results are marked in **Red**, second-best results are underlined. We report AUC and F1 scores here (↑).

Method	Bird lat, long → State	Income education → education_num	Housing latitude, longitude → CA	Average
TVAE	15.75±0.63	18.76±0.47	14.14±0.53	16.22±0.54
CTGAN	21.09±0.70	20.64±0.49	32.91±0.72	24.88±0.64
CTABGAN+	16.78±0.65	14.35±0.43	17.15±0.57	16.09±0.55
TabSyn	4.23±0.35	0.00±0.01	8.30±0.42	4.18±0.26
GReaT	1.98±0.24	0.15±0.05	3.51±0.28	1.88±0.19
GraDe	0.64±0.14	0.08±0.04	2.64±0.24	1.12±0.14
GraDe-Light	1.52±0.21	0.13±0.04	3.54±0.28	1.73±0.18

Table 5: Violation rates of intrinsic constraints in synthetic data (in %, a score closer to 0% is better) with 95% confidence intervals. The best results are marked in **Red**.

A.2 Intrinsic Constraint Fidelity

For LLM-based data synthesis methods, ensuring synthetic data is both diverse and preserves intrinsic constraints from original datasets is crucial (Rao et al., 2024, 2025; Chen et al., 2025; Lu et al., 2025; Qingsong et al., 2025; Han et al., 2025). Following Xu et al. (2024), we evaluate how well synthetic data preserves constraints that naturally exist in real-world datasets. Table 5 reports violation rates for three representative constraints: geographic coordinates matching state boundaries (Bird), education level matching education code (Income), and housing coordinates falling within California boundaries (Housing).

GraDe achieves the lowest average violation rate with 1.12%, significantly outperforming conventional generators. This superior performance stems from our explicit modeling of functional dependencies, which captures structural relationships between features. While GReaT also performs well (1.88% violations), our graph-guided approach provides more consistent constraint preservation. These results confirm that GraDe maintains both statistical properties and factual consistency, essential aspects of high-fidelity synthetic data.

A.3 Scaling to Larger Backbone Model

In principle, GraDe can be applied to any decoder-only language models. To investigate the impact of model scale on our approach, we conduct additional experiments using GPT-2 Medium (Radford et al., 2019) with 355 million parameters as the backbone model. Table 6 presents the MLE results for both GraDe and GraDe-Light with this larger model. These results complement our main findings and demonstrate the scalability of our approach across different model sizes.

A.4 Discriminator Measure

To further assess the *fidelity* of synthetic data beyond the correlation error histograms reported in §6, we employ a discriminator-based evaluation approach. We train a support vector machine (Cortes and Vapnik, 1995) with 5-fold cross-validation to distinguish between real and synthetic samples. For high-quality synthetic data, the discriminator accuracy should approach 50% (random chance), indicating the classifier cannot reliably differentiate between real and synthetic distributions.

Table 7 shows our generated samples effectively confuse classifiers trained on real data. While Tab-

Dataset		Original	TVAE	CTGAN	CTABGAN+	TabSyn	GReaT	**GraDe**	**GraDe-Light**
Bird (↑)	DT	99.97±0.02	70.59±0.00	94.28±0.17	66.46±0.21	99.75±0.06	99.63±0.07	99.76±0.02	99.78±0.00
	RF	100.00±0.00	79.87±0.46	99.09±0.08	75.54±0.11	99.97±0.12	100.00±0.00	100.00±0.00	99.96±0.02
	LR	100.00±0.00	87.04±0.00	100.00±0.00	73.99±0.00	100.00±0.00	98.29±0.00	100.00±0.00	98.35±0.00
Sick (↑)	DT	98.70±0.10	95.87±0.26	86.28±1.20	92.40±0.42	74.65±1.05	91.42±0.61	95.47±0.77	92.90±0.38
	RF	98.46±0.18	95.34±0.24	94.97±0.00	94.99±0.26	96.21±0.34	96.40±0.15	97.77±0.05	97.51±0.27
	LR	89.54±0.00	94.57±0.00	58.01±0.00	82.65±0.00	66.23±0.00	83.31±0.00	91.52±0.00	89.14±0.00
Income (↑)	DT	81.14±0.03	80.28±0.12	79.87±0.17	76.61±0.31	80.73±0.13	79.09±0.14	80.85±0.09	80.61±0.19
	RF	85.15±0.15	82.62±0.13	82.25±0.13	83.38±0.18	80.22±0.11	83.68±0.12	84.26±0.10	84.66±0.07
	LR	80.45±0.00	78.99±0.00	79.19±0.00	77.07±0.00	81.14±0.00	80.19±0.00	80.90±0.00	80.13±0.00
Diabetes (↑)	DT	74.68±1.30	72.21±0.49	59.48±0.88	62.99±1.23	75.06±2.08	65.97±0.66	76.62±0.88	72.21±1.12
	RF	74.94±0.88	75.97±0.71	57.01±1.50	64.81±1.50	75.32±0.92	67.40±0.76	73.90±0.95	75.32±0.71
	LR	69.48±0.00	73.38±0.00	57.14±0.00	72.08±0.00	70.78±0.00	67.53±0.00	69.48±0.00	68.18±0.00
Housing (↓)	DT	0.24±0.01	0.35±0.00	0.50±0.00	0.39±0.00	0.30±0.01	0.34±0.00	0.30±0.00	0.29±0.00
	RF	0.18±0.00	0.30±0.01	0.37±0.00	0.28±0.01	0.22±0.00	0.24±0.01	0.21±0.01	0.22±0.00
	LR	0.29±0.00	0.33±0.00	0.35±0.00	0.29±0.00	0.29±0.00	0.30±0.00	0.30±0.00	0.31±0.00

Table 6: Additional results of machine learning efficiency experiment. Here, GraDe and GraDe-Light use GPT-2 Medium as backbone model. The best results are marked in **Red**. Classification accuracy (↑) and regression mean absolute percentage error (↓) are reported.

Method	Bird	Sick	Income	Diabetes	Housing	Average
TVAE	74.95±0.25	76.22±2.26	67.27±1.21	67.59±6.58	60.37±0.97	69.28±2.26
CTGAN	59.90±0.98	81.54±2.66	62.58±0.81	85.50±3.22	62.89±0.77	70.48±1.69
CTABGAN+	63.77±1.35	64.35±2.13	58.49±0.42	60.75±2.47	60.41±0.44	61.55±1.36
TabSyn	49.32±0.37	54.82±2.74	55.41±0.85	45.36±2.79	50.48±0.71	51.08±1.49
GReaT	53.92±0.27	65.48±3.85	65.15±0.67	68.40±3.87	58.14±1.35	62.22±2.00
GraDe	52.33±0.67	62.33±1.62	59.49±0.85	59.28±2.37	56.54±0.80	57.99±1.26
GraDe-Light	53.43±0.74	63.44±2.66	64.64±0.90	66.45±2.61	58.40±0.47	61.27±1.48

Table 7: Discriminator measure (accuracy in %, a score closer to 50% is better) with 95% confidence intervals. The best results are marked in **Red**, second-best results are underlined.

Syn achieves closest to ideal performance at nearly 50% accuracy, our method outperforms other baselines. TabSyn's advantage likely stems from its score-based latent space guidance directing generation toward high-probability regions that closely mimic the original distribution. However, this close approximation may raise privacy concerns due to significant feature overlap between synthetic and real samples.

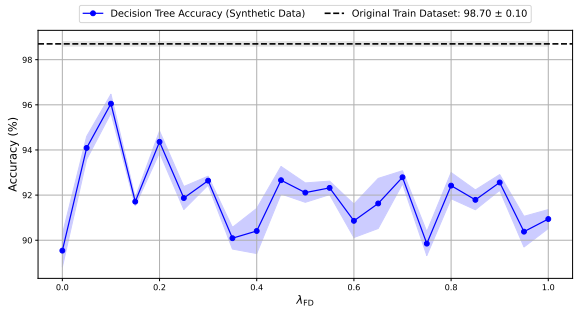

Figure 5: Impact of functional dependency alignment weight λ_{FD} on synthetic data quality, measured by classification accuracy on the **Sick** dataset.

A.5 Impact of FD Alignment Loss

In ablation study we observe that removing FD alignment loss has the most significant impact on medical datasets like Sick (§6.3). This is likely because such datasets contain many numerous numerical features and semantically ambiguous feature names (such as "T3", "TT4" in Sick), making it difficult for language models to capture dependencies implicitly without structural guidance.

To better understand this effect, we investigate how different FD alignment loss weights λ_{FD} (as given in Eq. 12) affect performance on challenging datasets. We train GraDe with varying λ_{FD} values on Sick and evaluate synthetic data quality using decision tree classifiers. Figure 5 shows the results, with the dashed line representing models trained on original data. The identified best value for λ_{FD} in Sick is 0.1.

A.6 Impact of Sparsity Constraint on High-Dimensional Data

To further validate the effectiveness of our sparsity constraint in more challenging scenarios, we

conducted an ablation study on the UCI Myocardial Infarction Complications (MIC) dataset (Golovenkin et al., 2020), which contains 124 features for predicting patient complications. This high-dimensional setting provides a more comprehensive evaluation of the sparsity constraint's impact compared to our initial experiments with 5–30 features.

We generated synthetic data under four different configurations and evaluated performance using decision tree (DT) classifiers, with results averaged across five random seeds. Table 8 presents the classification accuracy for each configuration.

Method	DT Accuracy (%) ↑
GraDe (full)	**89.41**
GraDe w/o Sparsity	86.18
GraDe w/o FDs	87.35
w/o both	84.41

Table 8: Ablation study on sparsity constraint effectiveness in high-dimensional data (124 features). Results show DT accuracy averaged over 5 random seeds.

The results confirm that the sparsity constraint demonstrates notably stronger effectiveness in this higher-dimensional setting, with a performance improvement of 3.23 percentage points compared to the variant without sparsity. Both sparsity and FD constraints prove complementary, with the FD constraint itself acting as an additional form of sparsity by selectively emphasizing essential dependencies. This validates our hypothesis that larger, more complex datasets better showcase the advantages of our constraint-guided approach.

B Implementation Details

B.1 Models and Evaluation

Models All models are implemented in PyTorch (Paszke et al., 2019) with pretrained language models from Huggingface (Wolf et al., 2020) as our foundations. We modified their source codes to integrate our graph-guided attention modules. For baselines, we use hyperparameters recommended by their respective authors. All experiments were conducted on an NVIDIA A100 GPU with 80GB memory.

Hyperparameter Settings We maintain consistent hyperparameter configurations across all datasets for both GraDe and GraDe-Light. We use the AdamW optimizer (Loshchilov and Hutter, 2019) with a learning rate of 5×10^{-5}. Depending on the GPU memory limitations, we use a fixed batch size of 64. The sparsity and functional dependency regularization weights (λ_{sparse} and λ_{FD}) are set to 0.001 and 0.1 respectively. For the sampling step, we set the temperature to 0.7 and the nucleus sampling parameter to 0.95 for all experiments and datasets. And following Borisov et al. (2023), we use regular expressions to convert generated text back to a tabular format (Aho, 1990).

DT	**RF**		**LR**	**SVM**
max_depth	max_depth	n_estimators	max_iter	kernel
20	20	100	500	linear

Table 9: Hyperparameter settings of the classification and regression models used in our experiments.

Evaluation Settings For the machine learning efficiency (§6, §A.1, §A.5 and §A.6), low-data regimes (§6.2), ablation study (§6.3) and discriminator experiments (§A.4) 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). Table 9 summarizes the hyperparameter settings used for all models in our experiments. For the scaling experiments (§A.1) on the imbalanced Loan dataset, we additionally enable class weighting for all classifiers to properly account for the class imbalance.

B.2 Parameter Counts and Efficiency

To enhance parameter efficiency, we introduce GraDe-Light, a lightweight variant that updates only the dynamic graph-guided attention modules while freezing all other parameters. As shown in Table 10, this approach substantially reduces the number of trainable parameters by 77.2% for GPT-2 and 71.6% for GPT-2 Medium. Despite this significant reduction, GraDe-Light maintains competitive performance across most datasets, offering a practical solution for computationally constrained environments while preserving the core benefits of our approach.

B.3 Datasets

We evaluate our approach on six real-world datasets, with five used in our main experiments and two used for our additional experiments. Bellows are the introductions for each dataset:

- **Bird**: The dataset contains geographic coordinates, official state birds, and latitude zones for

Method	Model	#Parameters	#Trainable
GraDe	GPT-2	124.5M	124.5M
GraDe-Light	GPT-2	124.5M	28.4M
GraDe	GPT-2 Medium	355.0M	355.0M
GraDe-Light	GPT-2 Medium	355.0M	100.9M

Table 10: Parameter comparison between GraDe and GraDe-Light across different foundation models. #Parameters and #Trainable denote the number of total parameters and the trainable parameters. Parameter counts are approximate and include the dynamic graph-guided attention modules.

all 51 US states. The task is to predict the official state bird based on location data.

- **Sick**: The dataset contains patient records with demographic information, medical history, and thyroid hormone measurements. The task is to predict thyroid disease status based on clinical features.
- **Income**: The dataset contains demographic and employment information. The task is to predict whether an individual's income exceeds $50,000 based on personal attributes.
- **Diabetes**: The dataset collected by the National Institute of Diabetes and Digestive and Kidney Diseases (Smith et al., 1988), contains clinical measurements of women of Pima Indian heritage. The task is to predict whether a patient has diabetes based on health indicators.
- **Housing**: The dataset contains district-level housing and demographic information for California. The task is to predict the median house value of each district.
- **Loan**: The dataset contains personal financial, demographic, and employment information about individuals from various cities in India. The task is to predict the loan risk based on personal attributes.
- **MIC**: The dataset contains clinical measurements from myocardial infarction patients during hospitalization. The task is to predict complications of myocardial infarction based on patient information collected at admission and on the third day of treatment.

For all datasets, we use an 80%/20% train-test split for model training and evaluation. Table 11 provides comprehensive statistics for all datasets.

B.4 Functional Dependency Extraction

To incorporate functional dependencies as structural priors into GraDe, we employ well-established algorithms from data profiling literature. Specifically, we adopt HyFD (Papenbrock and Naumann, 2016) as our primary extraction method, supplemented by TANE (Huhtala et al., 1999) as a verification reference.

TANE (Huhtala et al., 1999): A classical, lattice-based FD discovery algorithm that systematically enumerates all attribute combinations using a breadth-first, level-wise traversal. It computes partitions of tuples and exhaustively verifies all candidate dependencies, guaranteeing completeness and soundness. While theoretically rigorous, TANE's computational complexity grows exponentially with the number of attributes, making it impractical for datasets with many features or large-scale data.

HyFD (Papenbrock and Naumann, 2016): A more recent, hybrid approach designed explicitly to handle large-scale, real-world datasets efficiently. The algorithm proceeds in two key phases:

In the first phase (the *candidate generation phase*), HyFD employs a sampling-based method that examines subsets of the dataset to rapidly generate a comprehensive list of potential FD candidates. Specifically, it selects random pairs of data tuples, constructs *difference sets* (capturing attribute differences between tuple pairs), and leverages these sets to prune impossible or improbable FD candidates efficiently. This sampling-driven strategy drastically reduces computational cost compared to exhaustive enumeration.

The second phase (the *validation phase*) systematically verifies the previously identified FD candidates by constructing partition-based equivalence classes over the full dataset. HyFD efficiently refines and prunes these partitions to confirm or discard each candidate FD rigorously. This hybrid strategy effectively balances computational efficiency and completeness, supporting both exact and approximate functional dependencies, and ensures high reliability across varied data conditions, including noise or sparsity.

Practical Choice for Structural Guidance Given its strong trade-off between completeness and efficiency, we use HyFD throughout our experiments to extract structural priors. Its scalability and robustness to noise make it a practical choice

Dataset	Domain	#Samples	#Features	Task	#Classes	#Train [Pos%]	#Test [Pos%]
Bird (Evans-Bye, 2015)	Ecology	16,087	5	Classification	30	-	-
Sick (Dua and Graff, 2017)	Medical	3772	30	Classification	2	6.4%	5.0%
Income (Dua and Graff, 2017)	Social	32,561	15	Classification	2	24.2%	23.6%
Diabetes (From Kaggle[1])	Medical	768	9	Classification	2	34.7%	35.7%
Housing (Pace and Barry, 1997)	Real Estate	20,640	10	Regression	-	-	-
Loan (From Kaggle[2])	Financial	252,000	13	Classification	2	12.3%	12.4%
MIC (Golovenkin et al., 2020)	Medical	1,700	124	Classification	2	3.16%	3.24%

[1] https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database
[2] https://www.kaggle.com/datasets/subhamjain/loan-prediction-based-on-customer-behavior

Table 11: Statistics of the real-world datasets used in our experiments. The first five datasets are used for our main experiments. The shaded Loan and MIC datasets are used for our additional experiments (§A.1, §A.6). #Samples, #Features and #Classes denote the numbers of samples, features and classes in tabular datasets, respectively. For binary classification tasks, we report the percentage of positive samples in both training and test sets to highlight class imbalance.

for real-world applications where exhaustive enumeration is infeasible.

Challenges of Automated FD Extraction One potential concern with using automatically extracted functional dependencies is the risk that these structures may be incomplete or dataset-specific, raising the question of whether they can reliably guide modeling without manual verification. However, our approach uses extracted FDs not as absolute truth but as helpful structural hints from the data itself. Algorithms like HyFD identify statistically supported patterns that enhance model performance without requiring expert input. GraDe's design acknowledges that these dependencies might be imperfect; during training, the model naturally learns which dependencies are reliable and which are not, adjusting their influence accordingly. This makes our approach practical for real-world applications where manual verification would be prohibitively time-consuming.

Robustness Through Complementary Modeling GraDe achieves reliability through two complementary strategies. First, it employs a soft constraint formulation that allows the model to flexibly incorporate dependencies based on their observed reliability. This means GraDe naturally reduces the influence of noisy or incorrect dependencies while benefiting from valid ones. Second, our method combines two sources of structural information: statistical patterns from extracted FDs and semantic understanding from the pretrained LLM. This creates *natural resilience*: when extracted dependencies have flaws, the LLM's language understanding fills the gaps; when language patterns are unclear, the explicit dependencies provide guidance. This balanced approach ensures GraDe works effectively without expert intervention while being robust to imperfections in the dependency extraction process.

We provide the complete lists of FDs identified and utilized in our experiments here:

Bird

```
['lon', 'lat'] -> ['state_code', 'bird']
['lat'] -> ['lat_zone']
['state_code'] -> ['bird']
```

Sick

```
['TSH'] -> ['TSH_measured', 'TBG_measured',
'TBG']
['TSH', 'TT4'] -> ['hypopituitary']
['TSH', 'TT4', 'FTI'] -> ['I131_treatment',
'psych']
['TSH', 'TT4', 'FTI', 'T4U',
'query_on_thyroxine'] -> ['Class']
['TSH', 'TT4', 'FTI', 'age'] -> ['T4U']
['TSH', 'TT4', 'FTI', 'tumor', 'age', 'sex']
-> ['referral_source']
['TSH', 'TT4', 'FTI', 'age', 'T3_measured',
'sex'] -> ['referral_source']
['TSH', 'TT4', 'FTI', 'T3'] ->
['thyroid_surgery', 'lithium']
['TSH', 'TT4', 'FTI', 'T3',
'query_on_thyroxine'] -> ['T4U']
['TSH', 'TT4', 'FTI', 'referral_source'] ->
['thyroid_surgery', 'lithium']
['TSH', 'TT4', 'FTI', 'T3_measured',
'referral_source', 'sex',
'query_on_thyroxine'] -> ['T4U']
['TSH', 'TT4', 'FTI', 'referral_source',
'sex', 'query_on_thyroxine', 'Class'] ->
['T4U']
['TSH', 'TT4', 'FTI', 'sex'] ->
['thyroid_surgery', 'lithium']
['TSH', 'TT4', 'FTI', 'on_thyroxine'] ->
['thyroid_surgery']
['TSH', 'TT4', 'FTI', 'T3_measured'] ->
['thyroid_surgery', 'Class']
['TSH', 'TT4', 'T4U'] -> ['I131_treatment',
'lithium']
['TSH', 'TT4', 'T4U', 'age'] -> ['FTI',
'psych', 'FTI_measured', 'Class']
['TSH', 'TT4', 'T4U', 'age', 'tumor', 'sex']
-> ['referral_source']
['TSH', 'TT4', 'T4U', 'age', 'T3_measured',
'sex'] -> ['referral_source']
['TSH', 'TT4', 'T4U', 'T3'] ->
```

['query_on_thyroxine', 'thyroid_surgery', 'psych']
['TSH', 'TT4', 'T4U', 'referral_source'] -> ['psych']
['TSH', 'TT4', 'T4U', 'referral_source', 'query_hypothyroid'] -> ['thyroid_surgery']
['TSH', 'TT4', 'T4U', 'sex'] -> ['thyroid_surgery']
['TSH', 'TT4', 'T4U', 'sex', 'query_on_thyroxine'] -> ['Class']
['TSH', 'TT4', 'T4U', 'sex', 'T3_measured'] -> ['Class']
['TSH', 'TT4', 'T4U', 'on_thyroxine', 'query_hypothyroid'] -> ['thyroid_surgery']
['TSH', 'TT4', 'T4U', 'query_on_thyroxine', 'sick'] -> ['Class']
['TSH', 'TT4', 'T4U', 'sick', 'T3_measured'] -> ['Class']
['TSH', 'TT4', 'goitre', 'T4U'] -> ['psych']
['TSH', 'TT4', 'T4U', 'T3_measured'] -> ['query_on_thyroxine', 'thyroid_surgery', 'FTI_measured']
['TSH', 'TT4', 'T4U', 'Class'] -> ['query_on_thyroxine']
['TSH', 'TT4', 'age'] -> ['query_on_thyroxine', 'I131_treatment', 'lithium']
['TSH', 'TT4', 'age', 'T3'] -> ['psych', 'T4U_measured', 'FTI_measured']
['TSH', 'TT4', 'age', 'T3', 'sex', 'sick'] -> ['referral_source']
['TSH', 'TT4', 'age', 'referral_source'] -> ['psych']
['TSH', 'TT4', 'age', 'referral_source', 'thyroid_surgery'] -> ['T4U_measured', 'FTI_measured']
['TSH', 'TT4', 'age', 'referral_source', 'query_hypothyroid'] -> ['T4U_measured', 'FTI_measured']
['TSH', 'TT4', 'age', 'referral_source', 'Class'] -> ['T4U_measured', 'FTI_measured']
['TSH', 'TT4', 'age', 'sex', 'thyroid_surgery'] -> ['T4U_measured', 'FTI_measured']
['TSH', 'TT4', 'age', 'sex', 'query_hypothyroid'] -> ['T4U_measured', 'FTI_measured']
['TSH', 'TT4', 'age', 'sex', 'Class'] -> ['T4U_measured', 'FTI_measured']
['TSH', 'TT4', 'age', 'thyroid_surgery'] -> ['Class']
['TSH', 'TT4', 'age', 'query_hypothyroid'] -> ['thyroid_surgery', 'Class']
['TSH', 'TT4', 'age', 'T4U_measured'] -> ['thyroid_surgery', 'FTI_measured', 'Class']
['TSH', 'TT4', 'FTI_measured', 'age'] -> ['thyroid_surgery', 'Class']
['TSH', 'TT4', 'Class', 'age'] -> ['thyroid_surgery']
['TSH', 'TT4', 'T3'] -> ['Class']
['TSH', 'TT4', 'T3', 'referral_source'] -> ['lithium']
['TSH', 'TT4', 'T3', 'referral_source', 'sex'] -> ['psych']
['TSH', 'TT4', 'T3', 'referral_source', 'query_hypothyroid'] -> ['thyroid_surgery']
['TSH', 'TT4', 'T3', 'referral_source', 'tumor'] -> ['thyroid_surgery']
['TSH', 'TT4', 'T3', 'referral_source', 'T4U_measured'] -> ['psych']
['TSH', 'TT4', 'T3', 'referral_source', 'FTI_measured'] -> ['psych']
['TSH', 'TT4', 'T3', 'sex'] -> ['thyroid_surgery', 'I131_treatment', 'lithium', 'T4U_measured', 'FTI_measured']
['TSH', 'TT4', 'T3', 'sex', 'on_antithyroid_medication'] -> ['psych']
['TSH', 'TT4', 'T3', 'on_thyroxine'] -> ['thyroid_surgery']
['TSH', 'TT4', 'referral_source', 'sex'] -> ['lithium']
['TSH', 'TT4', 'T3_measured', 'referral_source', 'sex', 'query_hypothyroid'] -> ['thyroid_surgery']
['TSH', 'TT4', 'T3_measured', 'referral_source', 'sex', 'T4U_measured'] -> ['thyroid_surgery']
['TSH', 'TT4', 'T3_measured', 'referral_source', 'sex', 'FTI_measured'] -> ['thyroid_surgery']
['TSH', 'TT4', 'T3_measured', 'referral_source', 'sex', 'Class'] -> ['thyroid_surgery']
['TSH', 'TT4', 'query_hyperthyroid', 'referral_source', 'T3_measured', 'on_thyroxine', 'T4U_measured'] -> ['thyroid_surgery']
['TSH', 'TT4', 'query_hyperthyroid', 'referral_source', 'T3_measured', 'on_thyroxine', 'FTI_measured'] -> ['thyroid_surgery']
['TSH', 'TT4', 'query_hyperthyroid', 'referral_source', 'T3_measured', 'on_thyroxine', 'Class'] -> ['thyroid_surgery']
['TSH', 'TT4', 'query_hyperthyroid', 'referral_source', 'T3_measured', 'query_hypothyroid'] -> ['thyroid_surgery']
['TSH', 'TT4', 'query_hyperthyroid', 'tumor', 'referral_source', 'T3_measured', 'T4U_measured'] -> ['thyroid_surgery']
['TSH', 'TT4', 'query_hyperthyroid', 'tumor', 'referral_source', 'T3_measured', 'FTI_measured'] -> ['thyroid_surgery']
['TSH', 'TT4', 'query_hyperthyroid', 'tumor', 'referral_source', 'T3_measured', 'Class'] -> ['thyroid_surgery']
['TSH', 'TT4', 'psych', 'sex'] -> ['lithium']
['TSH', 'TT4', 'T4U_measured', 'T3_measured'] -> ['FTI_measured']
['TSH', 'TT4', 'FTI_measured'] -> ['T4U_measured']
['TSH', 'FTI'] -> ['hypopituitary']
['TSH', 'FTI', 'T4U'] -> ['lithium']
['TSH', 'FTI', 'T4U', 'age'] -> ['query_on_thyroxine', 'psych', 'Class']
['TSH', 'FTI', 'T4U', 'age', 'T3', 'TT4_measured'] -> ['TT4']
['TSH', 'FTI', 'T4U', 'age', 'tumor', 'sex', 'TT4_measured'] -> ['referral_source']
['TSH', 'FTI', 'T4U', 'age', 'T3_measured', 'sex', 'TT4_measured'] -> ['referral_source']
['TSH', 'FTI', 'T4U', 'age', 'sex', 'TT4_measured'] -> ['TT4']
['TSH', 'FTI', 'T4U', 'age', 'on_thyroxine', 'TT4_measured'] -> ['TT4']
['TSH', 'FTI', 'T4U', 'age', 'T3_measured', 'TT4_measured'] -> ['TT4']
['TSH', 'FTI', 'T4U', 'T3'] -> ['query_on_thyroxine', 'thyroid_surgery', 'psych', 'Class']
['TSH', 'FTI', 'T4U', 'referral_source'] -> ['psych']
['TSH', 'FTI', 'T4U', 'referral_source', 'query_hyperthyroid'] -> ['thyroid_surgery']
['TSH', 'FTI', 'T4U', 'referral_source', 'T3_measured'] -> ['query_on_thyroxine']
['TSH', 'FTI', 'T4U', 'referral_source', 'Class'] -> ['query_on_thyroxine']
['TSH', 'FTI', 'T4U', 'sex', 'query_hyperthyroid'] -> ['thyroid_surgery']
['TSH', 'FTI', 'T4U', 'on_thyroxine', 'query_hyperthyroid'] -> ['thyroid_surgery']
['TSH', 'FTI', 'T4U', 'query_hyperthyroid', 'T3_measured'] -> ['query_on_thyroxine', 'thyroid_surgery']
['TSH', 'FTI', 'T4U', 'query_hyperthyroid', 'Class'] -> ['query_on_thyroxine']
['TSH', 'FTI', 'T4U', 'psych', 'T3_measured'] -> ['query_on_thyroxine']
['TSH', 'FTI', 'T4U', 'psych', 'Class'] -> ['query_on_thyroxine']
['TSH', 'FTI', 'age'] -> ['lithium', 'T4U_measured']
['TSH', 'FTI', 'age', 'T3'] -> ['psych']
['TSH', 'FTI', 'age', 'T3', 'sex'] -> ['referral_source']
['TSH', 'FTI', 'age', 'T3', 'sex', 'on_thyroxine'] -> ['T4U']

['TSH', 'FTI', 'age', 'T3', 'sex', 'on_thyroxine', 'TT4_measured'] -> ['TT4']
['TSH', 'FTI', 'age', 'T3', 'sex', 'on_antithyroid_medication'] -> ['T4U']
['TSH', 'FTI', 'age', 'T3', 'sex', 'TT4_measured', 'on_antithyroid_medication'] -> ['TT4']
['TSH', 'FTI', 'age', 'T3', 'sex', 'query_hypothyroid'] -> ['T4U']
['TSH', 'FTI', 'age', 'T3', 'sex', 'TT4_measured', 'query_hypothyroid'] -> ['TT4']
['TSH', 'query_hyperthyroid', 'FTI', 'age', 'T3', 'sex'] -> ['T4U']
['TSH', 'query_hyperthyroid', 'FTI', 'age', 'T3', 'sex', 'TT4_measured'] -> ['TT4']
['TSH', 'FTI', 'goitre', 'age', 'T3', 'on_thyroxine'] -> ['T4U']
['TSH', 'FTI', 'goitre', 'age', 'T3', 'on_thyroxine', 'TT4_measured'] -> ['TT4']
['TSH', 'FTI', 'goitre', 'age', 'T3', 'on_antithyroid_medication'] -> ['T4U']
['TSH', 'FTI', 'goitre', 'age', 'T3', 'TT4_measured', 'on_antithyroid_medication'] -> ['TT4']
['TSH', 'FTI', 'goitre', 'age', 'T3', 'query_hypothyroid'] -> ['T4U']
['TSH', 'FTI', 'goitre', 'age', 'T3', 'TT4_measured', 'query_hypothyroid'] -> ['TT4']
['TSH', 'query_hyperthyroid', 'FTI', 'goitre', 'age', 'T3'] -> ['T4U']
['TSH', 'query_hyperthyroid', 'FTI', 'goitre', 'age', 'T3', 'TT4_measured'] -> ['TT4']
['TSH', 'FTI', 'age', 'referral_source'] -> ['psych', 'Class']
['TSH', 'FTI', 'age', 'referral_source', 'sex'] -> ['I131_treatment']
['TSH', 'FTI', 'age', 'referral_source', 'on_thyroxine'] -> ['I131_treatment']
['TSH', 'FTI', 'age', 'referral_source', 'TT4_measured'] -> ['I131_treatment']
['TSH', 'FTI', 'age', 'sex'] -> ['query_on_thyroxine', 'Class']
['TSH', 'FTI', 'age', 'sex', 'on_thyroxine'] -> ['I131_treatment']
['TSH', 'FTI', 'age', 'sex', 'T3_measured'] -> ['psych']
['TSH', 'FTI', 'age', 'on_thyroxine', 'query_hypothyroid'] -> ['I131_treatment']
['TSH', 'FTI', 'age', 'on_thyroxine', 'query_hyperthyroid'] -> ['I131_treatment']
['TSH', 'FTI', 'age', 'pregnant', 'query_hyperthyroid'] -> ['query_on_thyroxine']
['TSH', 'FTI', 'age', 'pregnant', 'tumor'] -> ['query_on_thyroxine']
['TSH', 'FTI', 'age', 'T3_measured'] -> ['query_on_thyroxine']
['TSH', 'FTI', 'T3'] -> ['I131_treatment']
['TSH', 'FTI', 'T3', 'referral_source'] -> ['lithium', 'Class']
['TSH', 'FTI', 'T3', 'referral_source', 'on_thyroxine'] -> ['thyroid_surgery']
['TSH', 'FTI', 'T3', 'on_thyroxine', 'sick'] -> ['thyroid_surgery']
['TSH', 'query_hyperthyroid', 'FTI', 'T3'] -> ['Class']
['TSH', 'FTI', 'referral_source', 'thyroid_surgery', 'query_hyperthyroid'] -> ['lithium']
['TSH', 'TT4_measured', 'FTI', 'referral_source'] -> ['T4U_measured']
['TSH', 'FTI', 'Class', 'referral_source'] -> ['lithium']
['TSH', 'TT4_measured', 'FTI', 'sex'] -> ['T4U_measured']
['TSH', 'FTI', 'query_hypothyroid', 'goitre', 'TT4_measured'] -> ['T4U_measured']
['TSH', 'FTI', 'goitre', 'TT4_measured', 'Class'] -> ['T4U_measured']
['TSH', 'T4U', 'age'] -> ['thyroid_surgery', 'lithium']
['TSH', 'T4U', 'age', 'T3'] -> ['I131_treatment']
['TSH', 'T4U', 'age', 'T3', 'sex'] -> ['referral_source', 'psych']
['TSH', 'T4U', 'age', 'referral_source'] -> ['FTI_measured', 'Class']
['TSH', 'T4U', 'age', 'referral_source', 'sex', 'on_thyroxine'] -> ['psych']
['TSH', 'T4U', 'age', 'referral_source', 'sex', 'sick'] -> ['psych']
['TSH', 'query_hyperthyroid', 'T4U', 'age', 'referral_source', 'sex'] -> ['psych']
['TSH', 'T4U', 'age', 'sex'] -> ['I131_treatment']
['TSH', 'T4U', 'age', 'sex', 'on_thyroxine', 'FTI_measured'] -> ['psych']
['TSH', 'T4U', 'age', 'sex', 'FTI_measured', 'sick'] -> ['psych']
['TSH', 'query_hyperthyroid', 'T4U', 'age', 'sex', 'FTI_measured'] -> ['psych']
['TSH', 'on_thyroxine', 'T4U', 'age'] -> ['I131_treatment']
['TSH', 'T4U', 'age', 'psych'] -> ['FTI_measured']
['TSH', 'TT4_measured', 'T4U', 'age'] -> ['I131_treatment']
['TSH', 'T4U', 'T3'] -> ['FTI_measured']
['TSH', 'T4U', 'T3', 'referral_source'] -> ['thyroid_surgery', 'lithium', 'psych']
['TSH', 'T4U', 'tumor', 'T3', 'referral_source', 'sex'] -> ['I131_treatment']
['TSH', 'T4U', 'T3', 'sex'] -> ['Class']
['TSH', 'T4U', 'tumor', 'T3', 'sex', 'on_thyroxine'] -> ['I131_treatment']
['TSH', 'T4U', 'T3', 'sex', 'thyroid_surgery'] -> ['lithium']
['TSH', 'T4U', 'T3', 'on_thyroxine', 'thyroid_surgery', 'query_hypothyroid'] -> ['lithium']
['TSH', 'query_hyperthyroid', 'T4U', 'T3'] -> ['Class']
['TSH', 'lithium', 'T4U', 'T3'] -> ['thyroid_surgery']
['TSH', 'T4U', 'referral_source'] -> ['hypopituitary']
['TSH', 'T4U', 'referral_source', 'sex', 'T3_measured'] -> ['FTI_measured']
['TSH', 'T4U', 'psych', 'referral_source'] -> ['lithium']
['TSH', 'T4U', 'referral_source', 'T3_measured', 'TT4_measured'] -> ['FTI_measured']
['TSH', 'T4U', 'T3_measured', 'psych', 'sex', 'on_thyroxine', 'thyroid_surgery', 'query_hypothyroid'] -> ['lithium']
['TSH', 'T4U', 'T3_measured', 'psych', 'sex', 'query_on_thyroxine'] -> ['FTI_measured']
['TSH', 'query_hyperthyroid', 'T4U', 'T3_measured', 'psych', 'sex'] -> ['FTI_measured']
['TSH', 'on_thyroxine', 'T4U'] -> ['hypopituitary']
['TSH', 'query_on_thyroxine', 'T4U'] -> ['hypopituitary']
['TSH', 'T4U', 'psych', 'T3_measured', 'TT4_measured', 'query_on_thyroxine'] -> ['FTI_measured']
['TSH', 'query_hyperthyroid', 'T4U', 'psych', 'T3_measured', 'TT4_measured'] -> ['FTI_measured']
['TSH', 'T4U', 'T3_measured'] -> ['hypopituitary']
['TSH', 'age'] -> ['hypopituitary']
['TSH', 'age', 'T3'] -> ['Class']
['TSH', 'age', 'T3', 'referral_source'] -> ['lithium', 'psych']
['TSH', 'age', 'T3', 'referral_source', 'on_thyroxine', 'query_hypothyroid'] -> ['thyroid_surgery']
['TSH', 'age', 'T3', 'referral_source', 'sick', 'query_hypothyroid'] -> ['thyroid_surgery']
['TSH', 'age', 'T3', 'sex'] -> ['thyroid_surgery', 'lithium']
['TSH', 'FTI_measured', 'age', 'T3'] ->

```
['T4U_measured']
['TSH', 'T4U_measured', 'age',
'referral_source'] -> ['FTI_measured']
['TSH', 'FTI_measured', 'age',
'referral_source'] -> ['T4U_measured']
['TSH', 'FTI_measured', 'age', 'sex'] ->
['T4U_measured']
['TSH', 'T4U_measured', 'age', 'psych'] ->
['FTI_measured']
['TSH', 'TT4_measured', 'FTI_measured',
'age'] -> ['T4U_measured']
['TSH', 'T3'] -> ['hypopituitary']
['TSH', 'T3', 'TT4_measured', 'FTI_measured',
'Class'] -> ['T4U_measured']
['TSH', 'T4U_measured', 'T3'] ->
['FTI_measured']
['TSH', 'referral_source', 'sex',
'T3_measured', 'T4U_measured'] ->
['FTI_measured']
['TSH', 'referral_source', 'T3_measured',
'TT4_measured', 'T4U_measured'] ->
['FTI_measured']
['TSH', 'T3_measured', 'psych', 'sex',
'query_on_thyroxine', 'T4U_measured'] ->
['FTI_measured']
['TSH', 'query_hyperthyroid', 'T3_measured',
'psych', 'sex', 'T4U_measured'] ->
['FTI_measured']
['TSH', 'psych', 'T3_measured',
'TT4_measured', 'query_on_thyroxine',
'T4U_measured'] -> ['FTI_measured']
['TSH', 'query_on_thyroxine', 'Class'] ->
['hypopituitary']
['TSH', 'query_hyperthyroid', 'psych',
'T3_measured', 'TT4_measured',
'T4U_measured'] -> ['FTI_measured']
['TT4'] -> ['TT4_measured', 'TBG_measured',
'TBG']
['TT4', 'FTI'] -> ['hypopituitary',
'T4U_measured']
['TT4', 'FTI', 'age'] ->
['query_on_thyroxine', 'thyroid_surgery',
'I131_treatment', 'lithium', 'Class']
['TT4', 'FTI', 'age', 'T3'] -> ['T4U',
'psych']
['TT4', 'FTI', 'age', 'T3',
'referral_source', 'sex', 'TSH_measured',
'query_hypothyroid'] -> ['TSH']
['TT4', 'FTI', 'age', 'T3_measured',
'referral_source', 'sex'] -> ['T4U']
['TT4', 'FTI', 'age', 'psych',
'referral_source', 'T3_measured'] -> ['T4U']
['TT4', 'FTI', 'age', 'sex'] -> ['psych']
['TT4', 'FTI', 'age', 'T3_measured', 'sex',
'sick'] -> ['T4U']
['TT4', 'FTI', 'age', 'psych', 'T3_measured',
'sick'] -> ['T4U']
['TT4', 'FTI', 'T3'] -> ['Class']
['TT4', 'FTI', 'T3', 'referral_source'] ->
['lithium']
['TT4', 'FTI', 'T3', 'referral_source',
'sex'] -> ['psych']
['TT4', 'FTI', 'T3', 'TSH_measured'] ->
['lithium']
['TT4', 'FTI', 'referral_source', 'sex',
'Class'] -> ['lithium']
['TT4', 'FTI', 'referral_source',
'on_thyroxine', 'Class', 'sick',
'I131_treatment'] -> ['lithium']
['query_hyperthyroid', 'TT4', 'FTI',
'referral_source', 'on_thyroxine', 'Class',
'I131_treatment'] -> ['lithium']
['TT4', 'FTI', 'referral_source',
'T3_measured', 'Class', 'sick',
'I131_treatment'] -> ['lithium']
['query_hyperthyroid', 'TT4', 'FTI',
'referral_source', 'T3_measured', 'Class',
'I131_treatment'] -> ['lithium']
['TT4', 'T4U'] -> ['hypopituitary']
['TT4', 'T4U', 'age'] -> ['thyroid_surgery',
'I131_treatment', 'lithium']
['TT4', 'T4U', 'age', 'T3'] -> ['FTI',
'query_on_thyroxine', 'psych']
['TT4', 'T4U', 'age', 'T3',
'referral_source', 'sex', 'TSH_measured',
'query_hypothyroid'] -> ['TSH']
['TT4', 'T4U', 'age', 'referral_source'] ->
['psych', 'Class']
['TT4', 'T4U', 'age', 'sex'] -> ['Class']
['on_thyroxine', 'TT4', 'T4U', 'age'] ->
['Class']
['TT4', 'T4U', 'age', 'query_hyperthyroid',
'T3_measured'] -> ['FTI_measured']
['TT4', 'T4U', 'age', 'TSH_measured'] ->
['FTI_measured']
['TT4', 'T4U', 'age', 'T3_measured'] ->
['query_on_thyroxine']
['TT4', 'T4U', 'T3'] -> ['Class']
['TT4', 'T4U', 'T3', 'referral_source'] ->
['lithium']
['TT4', 'T4U', 'T3', 'referral_source',
'sex'] -> ['psych']
['TT4', 'T4U', 'T3', 'referral_source',
'query_hyperthyroid'] -> ['psych']
['TT4', 'T4U', 'T3', 'sex'] -> ['lithium']
['TT4', 'T4U', 'T3', 'sex', 'on_thyroxine']
-> ['thyroid_surgery']
['TT4', 'T4U', 'T3', 'sex',
'query_hypothyroid'] -> ['thyroid_surgery']
['TT4', 'T4U', 'T3', 'query_on_thyroxine']
-> ['lithium']
['query_hyperthyroid', 'TT4', 'T4U', 'T3']
-> ['lithium']
['TT4', 'T4U', 'T3', 'TSH_measured'] ->
['query_on_thyroxine', 'lithium']
['TT4', 'T4U', 'psych', 'referral_source',
'sex', 'Class'] -> ['lithium']
['TT4', 'T4U', 'psych', 'referral_source',
'on_thyroxine', 'Class', 'sick',
'I131_treatment'] -> ['lithium']
['query_hyperthyroid', 'TT4', 'T4U', 'psych',
'referral_source', 'on_thyroxine', 'Class',
'I131_treatment'] -> ['lithium']
['TT4', 'T4U', 'psych', 'referral_source',
'T3_measured', 'Class', 'sick',
'I131_treatment'] -> ['lithium']
['query_hyperthyroid', 'TT4', 'T4U', 'psych',
'referral_source', 'T3_measured', 'Class',
'I131_treatment'] -> ['lithium']
['TT4', 'age'] -> ['hypopituitary']
['TT4', 'age', 'T3'] -> ['thyroid_surgery',
'Class']
['TT4', 'age', 'T3', 'referral_source'] ->
['psych']
['TT4', 'age', 'T3', 'referral_source',
'sex'] -> ['I131_treatment']
['TT4', 'age', 'T3', 'referral_source',
'sex', 'TSH_measured'] -> ['T4U_measured',
'FTI_measured']
['TT4', 'age', 'T3', 'sex'] ->
['query_on_thyroxine']
['TT4', 'age', 'T3', 'sex', 'on_thyroxine']
-> ['I131_treatment']
['TT4', 'age', 'T3', 'sex', 'T4U_measured']
-> ['I131_treatment']
['TT4', 'age', 'T3', 'sex', 'FTI_measured']
-> ['I131_treatment']
['TT4', 'age', 'T3', 'query_hypothyroid'] ->
['query_on_thyroxine']
['TT4', 'age', 'T3', 'TSH_measured'] ->
['query_on_thyroxine']
['TT4', 'age', 'T3_measured',
'referral_source', 'sex', 'on_thyroxine',
'thyroid_surgery'] -> ['I131_treatment']
['TT4', 'age', 'query_hyperthyroid',
'T3_measured', 'T4U_measured'] ->
['FTI_measured']
['TT4', 'age', 'TSH_measured',
'T4U_measured'] -> ['FTI_measured']
['TT4', 'FTI_measured', 'age'] ->
['T4U_measured']
['TT4', 'T3'] -> ['hypopituitary']
['TT4', 'FTI_measured', 'T3'] ->
['T4U_measured']
['TT4', 'query_on_thyroxine'] ->
['hypopituitary']
['FTI'] -> ['FTI_measured', 'TBG_measured',
'TBG']
['FTI', 'T4U'] -> ['hypopituitary']
['FTI', 'T4U', 'age'] -> ['lithium']
```

```
['FTI', 'T4U', 'age', 'T3'] ->
['I131_treatment']
['FTI', 'T4U', 'age', 'T3',
'referral_source', 'sex', 'on_thyroxine',
'query_hypothyroid', 'TSH_measured'] ->
['TSH']
['FTI', 'T4U', 'age', 'T3',
'referral_source', 'sex', 'on_thyroxine',
'TT4_measured'] -> ['TT4']
['FTI', 'T4U', 'age', 'referral_source',
'sex'] -> ['psych', 'Class']
['FTI', 'T4U', 'age', 'referral_source',
'sick'] -> ['psych', 'Class']
['FTI', 'T4U', 'age', 'referral_source',
'query_hypothyroid'] -> ['psych', 'Class']
['FTI', 'T4U', 'age', 'referral_source',
'query_hyperthyroid'] ->
['query_on_thyroxine']
['FTI', 'T4U', 'age', 'referral_source',
'psych'] -> ['Class']
['FTI', 'T4U', 'age', 'referral_source',
'T3_measured'] -> ['psych', 'Class']
['FTI', 'T4U', 'age', 'referral_source',
'TT4_measured'] -> ['psych', 'Class']
['FTI', 'T4U', 'age', 'referral_source',
'Class'] -> ['psych']
['FTI', 'T4U', 'age', 'sex',
'query_hyperthyroid'] ->
['query_on_thyroxine']
['on_thyroxine', 'FTI', 'T4U', 'age'] ->
['I131_treatment']
['FTI', 'T4U', 'age', 'query_hyperthyroid',
'psych'] -> ['query_on_thyroxine']
['FTI', 'T4U', 'age', 'query_hyperthyroid',
'TSH_measured'] -> ['query_on_thyroxine']
['FTI', 'T4U', 'age', 'query_hyperthyroid',
'T3_measured'] -> ['query_on_thyroxine']
['FTI', 'T4U', 'T3', 'referral_source'] ->
['lithium']
['FTI', 'T4U', 'T3', 'referral_source',
'sex', 'sick'] -> ['psych']
['FTI', 'T4U', 'T3', 'referral_source',
'sex', 'query_hypothyroid'] -> ['psych']
['FTI', 'T4U', 'T3', 'referral_source',
'sex', 'Class'] -> ['psych']
['FTI', 'T4U', 'T3', 'referral_source',
'sick'] -> ['Class']
['query_hyperthyroid', 'FTI', 'T4U', 'T3',
'referral_source', 'sick'] -> ['psych']
['FTI', 'T4U', 'T3', 'referral_source',
'query_hypothyroid'] -> ['Class']
['query_hyperthyroid', 'FTI', 'T4U', 'T3',
'referral_source', 'query_hypothyroid'] ->
['psych']
['query_hyperthyroid', 'FTI', 'T4U', 'T3',
'referral_source', 'Class'] -> ['psych']
['FTI', 'T4U', 'T3', 'referral_source',
'psych'] -> ['Class']
['FTI', 'T4U', 'T3', 'sex'] -> ['lithium']
['FTI', 'T4U', 'T3', 'sex', 'TSH_measured']
-> ['query_on_thyroxine']
['query_on_thyroxine', 'FTI', 'T4U', 'T3']
-> ['lithium']
['query_hyperthyroid', 'FTI', 'T4U', 'T3']
-> ['lithium']
['FTI', 'T4U', 'T3', 'query_hyperthyroid',
'TSH_measured'] -> ['query_on_thyroxine']
['FTI', 'T4U', 'T3', 'TSH_measured'] ->
['lithium']
['query_hyperthyroid', 'FTI', 'T4U', 'psych',
'referral_source', 'sex', 'Class'] ->
['lithium']
['query_hyperthyroid', 'FTI', 'T4U', 'psych',
'referral_source', 'on_thyroxine', 'Class',
'I131_treatment'] -> ['lithium']
['query_hyperthyroid', 'FTI', 'T4U', 'psych',
'referral_source', 'T3_measured', 'Class',
'I131_treatment'] -> ['lithium']
['FTI', 'age'] -> ['hypopituitary']
['FTI', 'age', 'T3'] ->
['query_on_thyroxine', 'thyroid_surgery',
'T4U_measured', 'Class']
['FTI', 'age', 'T3', 'referral_source'] ->
['I131_treatment']
['FTI', 'age', 'T3', 'referral_source',
'psych'] -> ['lithium']
['FTI', 'age', 'T3', 'sex'] ->
['I131_treatment']
['FTI', 'age', 'psych', 'referral_source',
'sex', 'T3_measured'] -> ['lithium']
['query_hyperthyroid', 'FTI', 'age',
'psych', 'referral_source', 'T3_measured'] ->
['lithium']
['FTI', 'sick', 'age', 'T3_measured'] ->
['T4U_measured']
['TT4_measured', 'FTI', 'sick', 'age'] ->
['T4U_measured']
['FTI', 'Class', 'age', 'T3_measured'] ->
['T4U_measured']
['TT4_measured', 'FTI', 'Class', 'age'] ->
['T4U_measured']
['FTI', 'T3'] -> ['hypopituitary']
['TT4_measured', 'FTI', 'T3'] ->
['T4U_measured']
['FTI', 'referral_source', 'sex'] ->
['hypopituitary']
['on_thyroxine', 'FTI', 'tumor',
'referral_source'] -> ['hypopituitary']
['FTI', 'Class', 'sex'] -> ['hypopituitary']
['on_thyroxine', 'FTI', 'Class'] ->
['hypopituitary']
['query_on_thyroxine', 'FTI'] ->
['hypopituitary']
['T4U'] -> ['T4U_measured', 'TBG_measured',
'TBG']
['T4U', 'age', 'T3'] -> ['FTI_measured']
['T4U', 'age', 'T3', 'referral_source'] ->
['lithium', 'psych']
['T4U', 'age', 'T3', 'sex'] -> ['lithium',
'Class']
['T4U', 'age', 'T3', 'sex', 'TSH_measured']
-> ['I131_treatment']
['T4U', 'age', 'T3', 'query_hypothyroid'] ->
['Class']
['T4U', 'age', 'referral_source'] ->
['hypopituitary']
['T4U', 'age', 'psych', 'referral_source',
'sex', 'query_hypothyroid'] -> ['lithium']
['query_hyperthyroid', 'T4U', 'age',
'T3_measured', 'referral_source', 'sex']
-> ['FTI_measured']
['T4U', 'age', 'referral_source', 'sex',
'TSH_measured'] -> ['FTI_measured']
['query_hyperthyroid', 'T4U', 'age',
'referral_source', 'T3_measured',
'TT4_measured'] -> ['FTI_measured']
['T4U', 'age', 'referral_source',
'TSH_measured', 'TT4_measured'] ->
['FTI_measured']
['T4U', 'age', 'sex'] -> ['hypopituitary']
['query_hyperthyroid', 'T4U', 'age', 'psych',
'T3_measured', 'sex'] -> ['FTI_measured']
['T4U', 'age', 'sex', 'psych',
'TSH_measured'] -> ['FTI_measured']
['query_on_thyroxine', 'T4U', 'age'] ->
['hypopituitary']
['T4U', 'age', 'TSH_measured'] ->
['hypopituitary']
['T4U', 'age', 'T3_measured'] ->
['hypopituitary']
['T4U', 'T3', 'referral_source',
'query_hypothyroid'] -> ['hypopituitary']
['T4U', 'T3', 'sex'] -> ['hypopituitary']
['query_on_thyroxine', 'T4U', 'T3'] ->
['hypopituitary']
['query_on_thyroxine', 'T4U',
'referral_source'] -> ['hypopituitary']
['query_on_thyroxine', 'T4U',
'TSH_measured'] -> ['hypopituitary']
['Class', 'T4U'] -> ['hypopituitary']
['age'] -> ['TBG_measured', 'TBG']
['age', 'T3'] -> ['hypopituitary']
['age', 'T3', 'referral_source',
'on_thyroxine', 'FTI_measured',
'thyroid_surgery'] -> ['T4U_measured']
['age', 'T3', 'sex', 'on_thyroxine',
'FTI_measured', 'thyroid_surgery'] ->
['T4U_measured']
['TT4_measured', 'FTI_measured', 'age',
'T3'] -> ['T4U_measured']
```

```
['T4U_measured', 'age', 'T3'] ->
['FTI_measured']
['age', 'referral_source', 'sex'] ->
['hypopituitary']
['age', 'referral_source', 'sex',
'TT4_measured', 'FTI_measured', 'sick']
-> ['T4U_measured']
['query_hyperthyroid', 'age', 'T3_measured',
'referral_source', 'sex', 'T4U_measured'] ->
['FTI_measured']
['age', 'referral_source', 'sex',
'TSH_measured', 'T4U_measured'] ->
['FTI_measured']
['age', 'referral_source', 'sex',
'TT4_measured', 'FTI_measured', 'Class']
-> ['T4U_measured']
['query_on_thyroxine', 'age',
'referral_source'] -> ['hypopituitary']
['query_hyperthyroid', 'age',
'referral_source', 'T3_measured',
'TT4_measured', 'T4U_measured'] ->
['FTI_measured']
['age', 'referral_source', 'TSH_measured',
'TT4_measured', 'T4U_measured'] ->
['FTI_measured']
['query_on_thyroxine', 'age', 'sex'] ->
['hypopituitary']
['query_hyperthyroid', 'age', 'psych',
'T3_measured', 'sex', 'T4U_measured'] ->
['FTI_measured']
['age', 'sex', 'psych', 'TSH_measured',
'T4U_measured'] -> ['FTI_measured']
['query_on_thyroxine', 'age',
'TSH_measured'] -> ['hypopituitary']
['Class', 'age'] -> ['hypopituitary']
['T3'] -> ['T3_measured', 'TBG_measured',
'TBG']
['Class', 'T3'] -> ['hypopituitary']
['referral_source'] -> ['TBG_measured',
'TBG']
['referral_source', 'sex',
'query_on_thyroxine', 'goitre', 'Class']
-> ['hypopituitary']
['referral_source', 'query_on_thyroxine',
'sick', 'goitre', 'Class'] ->
['hypopituitary']
['sex'] -> ['TBG_measured', 'TBG']
['sex', 'query_on_thyroxine', 'goitre',
'TSH_measured', 'Class'] -> ['hypopituitary']
['on_thyroxine'] -> ['TBG_measured', 'TBG']
['query_on_thyroxine'] -> ['TBG_measured',
'TBG']
['query_on_thyroxine', 'sick', 'goitre',
'TSH_measured', 'Class'] -> ['hypopituitary']
['on_antithyroid_medication'] ->
['TBG_measured', 'TBG']
['sick'] -> ['TBG_measured', 'TBG']
['pregnant'] -> ['TBG_measured', 'TBG']
['thyroid_surgery'] -> ['TBG_measured',
'TBG']
['I131_treatment'] -> ['TBG_measured',
'TBG']
['query_hypothyroid'] -> ['TBG_measured',
'TBG']
['query_hyperthyroid'] -> ['TBG_measured',
'TBG']
['lithium'] -> ['TBG_measured', 'TBG']
['goitre'] -> ['TBG_measured', 'TBG']
['tumor'] -> ['TBG_measured', 'TBG']
['hypopituitary'] -> ['TBG_measured', 'TBG']
['psych'] -> ['TBG_measured', 'TBG']
['TSH_measured'] -> ['TBG_measured', 'TBG']
['T3_measured'] -> ['TBG_measured', 'TBG']
['TT4_measured'] -> ['TBG_measured', 'TBG']
['T4U_measured'] -> ['TBG_measured', 'TBG']
['FTI_measured'] -> ['TBG_measured', 'TBG']
['Class'] -> ['TBG_measured', 'TBG']
['TBG_measured'] -> ['TBG']
['TBG'] -> ['TBG_measured']
```

Income

```
['fnlwgt', 'capital-gain', 'hours-per-week',
'capital-loss', 'education', 'workclass',
'relationship'] -> ['gender']
['fnlwgt', 'capital-gain', 'hours-per-week',
'capital-loss', 'education-num', 'workclass',
'relationship'] -> ['gender']
['fnlwgt', 'capital-gain', 'hours-per-week',
'age', 'education', 'occupation',
'workclass', 'relationship'] -> ['income']
['fnlwgt', 'capital-gain', 'hours-per-week',
'age', 'education-num', 'occupation',
'workclass', 'relationship'] -> ['income']
['fnlwgt', 'capital-gain', 'hours-per-week',
'age', 'occupation', 'relationship'] ->
['marital-status']
['fnlwgt', 'capital-gain', 'hours-per-week',
'education', 'workclass', 'relationship',
'race'] -> ['gender']
['fnlwgt', 'capital-gain', 'hours-per-week',
'education', 'workclass', 'relationship',
'income'] -> ['gender']
['fnlwgt', 'capital-gain', 'hours-per-week',
'education-num', 'workclass', 'relationship',
'race'] -> ['gender']
['fnlwgt', 'capital-gain', 'hours-per-week',
'education-num', 'workclass', 'relationship',
'income'] -> ['gender']
['fnlwgt', 'hours-per-week', 'age',
'education'] -> ['race']
['fnlwgt', 'hours-per-week', 'age',
'education', 'occupation', 'workclass']
-> ['capital-loss']
['fnlwgt', 'hours-per-week', 'age',
'education', 'occupation', 'relationship'] ->
['marital-status']
['fnlwgt', 'hours-per-week', 'age',
'education-num'] -> ['race']
['fnlwgt', 'hours-per-week', 'age',
'education-num', 'occupation', 'workclass']
-> ['capital-loss']
['fnlwgt', 'hours-per-week', 'age',
'education-num', 'occupation',
'relationship'] -> ['marital-status']
['fnlwgt', 'hours-per-week', 'age',
'occupation', 'workclass'] -> ['race']
['fnlwgt', 'hours-per-week', 'age',
'workclass', 'marital-status',
'relationship'] -> ['race']
['fnlwgt', 'hours-per-week',
'native-country', 'education',
'marital-status', 'relationship'] -> ['race']
['fnlwgt', 'hours-per-week',
'native-country', 'education-num',
'marital-status', 'relationship'] -> ['race']
['fnlwgt', 'hours-per-week',
'native-country', 'occupation', 'workclass',
'gender'] -> ['race']
['fnlwgt', 'hours-per-week',
'native-country', 'occupation',
'marital-status'] -> ['race']
['fnlwgt', 'hours-per-week',
'native-country', 'occupation',
'relationship'] -> ['race']
['fnlwgt', 'hours-per-week',
'native-country', 'occupation', 'gender',
'income'] -> ['race']
['fnlwgt', 'hours-per-week', 'education',
'marital-status', 'relationship', 'income']
-> ['race']
['fnlwgt', 'hours-per-week', 'education-num',
'marital-status', 'relationship', 'income']
-> ['race']
['fnlwgt', 'occupation', 'hours-per-week',
'relationship'] -> ['gender']
['fnlwgt', 'hours-per-week', 'workclass',
'marital-status', 'relationship'] ->
['gender']
['fnlwgt', 'age', 'native-country',
'education'] -> ['race']
['fnlwgt', 'age', 'native-country',
'education-num'] -> ['race']
['fnlwgt', 'age', 'native-country',
'occupation', 'workclass'] -> ['race']
['fnlwgt', 'age', 'native-country',
```

```
'workclass', 'marital-status'] -> ['race']
['fnlwgt', 'marital-status', 'age',
'education'] -> ['race', 'gender']
['fnlwgt', 'relationship', 'age',
'education'] -> ['race']
['fnlwgt', 'age', 'education', 'income'] ->
['race']
['fnlwgt', 'marital-status', 'age',
'education-num'] -> ['race', 'gender']
['fnlwgt', 'relationship', 'age',
'education-num'] -> ['race']
['fnlwgt', 'age', 'income', 'education-num']
-> ['race']
['fnlwgt', 'occupation', 'marital-status',
'age'] -> ['race', 'gender']
['fnlwgt', 'occupation', 'relationship',
'age'] -> ['race']
['fnlwgt', 'occupation', 'age', 'income'] ->
['race']
['fnlwgt', 'age', 'workclass',
'marital-status', 'relationship', 'income']
-> ['race']
['fnlwgt', 'relationship', 'age'] ->
['gender']
['fnlwgt', 'native-country', 'education',
'occupation', 'gender'] -> ['race']
['fnlwgt', 'native-country', 'education',
'workclass', 'marital-status',
'relationship', 'gender'] -> ['race']
['fnlwgt', 'native-country', 'education-num',
'occupation', 'gender'] -> ['race']
['fnlwgt', 'native-country', 'education-num',
'workclass', 'marital-status',
'relationship', 'gender'] -> ['race']
['fnlwgt', 'education', 'occupation',
'workclass', 'relationship'] -> ['gender']
['fnlwgt', 'occupation', 'relationship',
'education'] -> ['race']
['fnlwgt', 'education', 'occupation',
'relationship', 'income'] -> ['gender']
['fnlwgt', 'education', 'workclass',
'marital-status', 'relationship', 'gender',
'income'] -> ['race']
['fnlwgt', 'education-num', 'occupation',
'workclass', 'relationship'] -> ['gender']
['fnlwgt', 'occupation', 'relationship',
'education-num'] -> ['race']
['fnlwgt', 'education-num', 'occupation',
'relationship', 'income'] -> ['gender']
['fnlwgt', 'education-num', 'workclass',
'marital-status', 'relationship', 'gender',
'income'] -> ['race']
['fnlwgt', 'occupation', 'marital-status',
'relationship'] -> ['gender']
['education'] -> ['education-num']
['education-num'] -> ['education']
```

Diabetes

```
['DiabetesPedigreeFunction', 'BMI',
'Insulin'] -> ['Glucose', 'Age',
'SkinThickness', 'BloodPressure',
'Pregnancies', 'Diabetes']
['DiabetesPedigreeFunction', 'BMI',
'Glucose'] -> ['Insulin', 'Age',
'SkinThickness', 'BloodPressure',
'Pregnancies', 'Diabetes']
['DiabetesPedigreeFunction', 'BMI', 'Age']
-> ['Insulin', 'Glucose', 'SkinThickness',
'BloodPressure', 'Pregnancies', 'Diabetes']
['DiabetesPedigreeFunction', 'BMI',
'SkinThickness'] -> ['Insulin', 'Glucose',
'Age', 'BloodPressure', 'Pregnancies',
'Diabetes']
['DiabetesPedigreeFunction', 'BMI',
'BloodPressure'] -> ['Insulin', 'Glucose',
'Age', 'SkinThickness', 'Pregnancies',
'Diabetes']
['DiabetesPedigreeFunction', 'BMI',
'Pregnancies'] -> ['Insulin', 'Glucose',
'Age', 'SkinThickness', 'BloodPressure',
'Diabetes']
['DiabetesPedigreeFunction', 'BMI',
'Diabetes'] -> ['Insulin', 'Glucose',
'Age', 'SkinThickness', 'BloodPressure',
'Pregnancies']
['DiabetesPedigreeFunction', 'Insulin',
'Glucose'] -> ['BMI', 'Age', 'SkinThickness',
'BloodPressure', 'Pregnancies', 'Diabetes']
['DiabetesPedigreeFunction', 'Insulin',
'Age', 'Pregnancies'] -> ['SkinThickness']
['DiabetesPedigreeFunction', 'Insulin',
'BloodPressure'] -> ['SkinThickness',
'Diabetes']
['DiabetesPedigreeFunction', 'Glucose',
'Age'] -> ['BMI', 'Insulin', 'SkinThickness',
'BloodPressure', 'Pregnancies', 'Diabetes']
['DiabetesPedigreeFunction', 'Glucose',
'SkinThickness'] -> ['BMI', 'Insulin', 'Age',
'BloodPressure', 'Pregnancies', 'Diabetes']
['DiabetesPedigreeFunction', 'Glucose',
'BloodPressure'] -> ['Diabetes']
['DiabetesPedigreeFunction', 'Glucose',
'Pregnancies'] -> ['BMI', 'Insulin', 'Age',
'SkinThickness', 'BloodPressure', 'Diabetes']
['DiabetesPedigreeFunction', 'Diabetes',
'Glucose'] -> ['BloodPressure']
['DiabetesPedigreeFunction', 'Age',
'SkinThickness'] -> ['Insulin']
['DiabetesPedigreeFunction', 'Age',
'BloodPressure'] -> ['BMI', 'Insulin',
'Glucose', 'SkinThickness', 'Pregnancies',
'Diabetes']
['DiabetesPedigreeFunction', 'Age',
'Pregnancies'] -> ['Diabetes']
['DiabetesPedigreeFunction', 'SkinThickness',
'BloodPressure'] -> ['Insulin', 'Diabetes']
['DiabetesPedigreeFunction', 'SkinThickness',
'Pregnancies'] -> ['Insulin', 'Diabetes']
['DiabetesPedigreeFunction', 'BloodPressure',
'Pregnancies'] -> ['Insulin',
'SkinThickness', 'Diabetes']
['BMI', 'Insulin', 'Glucose', 'Age'] ->
['DiabetesPedigreeFunction', 'SkinThickness',
'BloodPressure']
['BMI', 'Insulin', 'Glucose',
'BloodPressure'] ->
['DiabetesPedigreeFunction', 'Age',
'SkinThickness', 'Pregnancies']
['BMI', 'Insulin', 'Glucose', 'Pregnancies']
-> ['DiabetesPedigreeFunction', 'Age',
'SkinThickness', 'BloodPressure', 'Diabetes']
['BMI', 'Insulin', 'Age',
'SkinThickness', 'Pregnancies'] ->
['DiabetesPedigreeFunction', 'Glucose',
'BloodPressure']
['BMI', 'Glucose', 'Age'] -> ['Pregnancies',
'Diabetes']
['BMI', 'Glucose', 'Age', 'SkinThickness']
-> ['DiabetesPedigreeFunction',
'BloodPressure']
['BMI', 'Glucose', 'Age', 'BloodPressure']
-> ['DiabetesPedigreeFunction', 'Insulin',
'SkinThickness']
['BMI', 'Glucose', 'SkinThickness'] ->
['Insulin']
['BMI', 'Glucose', 'SkinThickness',
'BloodPressure'] ->
['DiabetesPedigreeFunction', 'Age',
'Pregnancies']
['BMI', 'Glucose',
'SkinThickness', 'Pregnancies'] ->
['DiabetesPedigreeFunction', 'Age',
'BloodPressure', 'Diabetes']
['Diabetes', 'BMI', 'Glucose',
'SkinThickness'] ->
['DiabetesPedigreeFunction', 'Age',
'BloodPressure', 'Pregnancies']
['BMI', 'Glucose', 'BloodPressure'] ->
['Diabetes']
['BMI', 'Glucose',
'BloodPressure', 'Pregnancies'] ->
['DiabetesPedigreeFunction', 'Insulin',
'Age', 'SkinThickness']
['Diabetes', 'BMI', 'Glucose',
'Pregnancies'] -> ['Age']
['BMI', 'Age', 'SkinThickness',
'BloodPressure'] -> ['Insulin', 'Diabetes']
['BMI', 'Age', 'SkinThickness',
```

```
'BloodPressure', 'Pregnancies'] ->
['DiabetesPedigreeFunction', 'Glucose']
['BMI', 'Age', 'SkinThickness',
'Pregnancies'] -> ['Diabetes']
['BMI', 'SkinThickness', 'BloodPressure',
'Pregnancies'] -> ['Insulin']
['Insulin', 'Glucose',
'Age', 'BloodPressure'] ->
['DiabetesPedigreeFunction', 'BMI',
'SkinThickness', 'Pregnancies', 'Diabetes']
['Insulin', 'Glucose', 'Age', 'Pregnancies']
-> ['Diabetes']
['Glucose', 'Age', 'SkinThickness',
'BloodPressure'] ->
['DiabetesPedigreeFunction', 'BMI',
'Insulin', 'Pregnancies', 'Diabetes']
['Glucose', 'Age', 'SkinThickness',
'Pregnancies'] -> ['Diabetes']
['Glucose', 'Age',
'BloodPressure', 'Pregnancies'] ->
['DiabetesPedigreeFunction', 'BMI',
'Insulin', 'SkinThickness', 'Diabetes']
['Glucose', 'SkinThickness', 'BloodPressure',
'Pregnancies'] -> ['Insulin']
['Glucose', 'SkinThickness', 'BloodPressure',
'Pregnancies', 'Diabetes'] ->
['DiabetesPedigreeFunction', 'BMI', 'Age']
```

Housing

```
['median_income', 'total_rooms',
'median_house_value'] -> ['population',
'total_bedrooms', 'households', 'latitude',
'longitude', 'housing_median_age',
'ocean_proximity']
['median_income', 'total_rooms',
'population'] -> ['median_house_value',
'total_bedrooms', 'households', 'latitude',
'longitude', 'housing_median_age',
'ocean_proximity']
['median_income', 'total_rooms',
'total_bedrooms'] -> ['median_house_value',
'population', 'households', 'latitude',
'longitude', 'housing_median_age',
'ocean_proximity']
['median_income', 'total_rooms',
'households'] -> ['median_house_value',
'population', 'total_bedrooms', 'latitude',
'longitude', 'housing_median_age',
'ocean_proximity']
['median_income', 'total_rooms', 'latitude']
-> ['median_house_value', 'population',
'total_bedrooms', 'households', 'longitude',
'housing_median_age', 'ocean_proximity']
['median_income', 'total_rooms',
'longitude'] -> ['median_house_value',
'population', 'total_bedrooms', 'households',
'latitude', 'housing_median_age',
'ocean_proximity']
['median_income', 'total_rooms',
'housing_median_age'] ->
['median_house_value', 'population',
'total_bedrooms', 'households', 'latitude',
'longitude', 'ocean_proximity']
['median_income', 'median_house_value',
'population'] -> ['ocean_proximity']
['median_income', 'median_house_value',
'population', 'latitude'] -> ['total_rooms',
'total_bedrooms', 'households', 'longitude',
'housing_median_age']
['median_income', 'median_house_value',
'total_bedrooms'] -> ['latitude',
'ocean_proximity']
['median_income', 'median_house_value',
'total_bedrooms', 'households'] ->
['total_rooms', 'population', 'longitude',
'housing_median_age']
['median_income', 'housing_median_age',
'median_house_value', 'total_bedrooms'] ->
['total_rooms', 'population', 'households',
'longitude']
['median_income', 'median_house_value',
'households'] -> ['ocean_proximity']
['median_income', 'housing_median_age',
'median_house_value', 'households']
-> ['total_rooms', 'population',
'total_bedrooms', 'latitude', 'longitude']
['median_income', 'median_house_value',
'latitude', 'longitude'] ->
['housing_median_age']
['median_income', 'population',
'total_bedrooms'] -> ['total_rooms',
'median_house_value', 'households',
'latitude', 'longitude',
'housing_median_age', 'ocean_proximity']
['median_income', 'population',
'households'] -> ['total_rooms',
'median_house_value', 'total_bedrooms',
'latitude', 'longitude',
'housing_median_age', 'ocean_proximity']
['median_income', 'population', 'latitude']
-> ['ocean_proximity']
['median_income', 'population', 'longitude']
-> ['total_rooms', 'median_house_value',
'total_bedrooms', 'households', 'latitude',
'housing_median_age', 'ocean_proximity']
['median_income', 'housing_median_age',
'population'] -> ['total_rooms',
'median_house_value', 'total_bedrooms',
'households', 'latitude', 'longitude',
'ocean_proximity']
['median_income', 'total_bedrooms',
'households'] -> ['ocean_proximity']
['median_income', 'housing_median_age',
'total_bedrooms', 'households'] ->
['total_rooms', 'median_house_value',
'population', 'latitude', 'longitude']
['median_income', 'total_bedrooms',
'latitude'] -> ['ocean_proximity']
['median_income', 'housing_median_age',
'total_bedrooms', 'latitude'] ->
['total_rooms', 'median_house_value',
'population', 'households', 'longitude']
['median_income', 'total_bedrooms',
'longitude'] -> ['total_rooms',
'median_house_value', 'population',
'households', 'latitude',
'housing_median_age', 'ocean_proximity']
['median_income', 'housing_median_age',
'total_bedrooms'] -> ['ocean_proximity']
['median_income', 'households', 'latitude']
-> ['total_rooms', 'median_house_value',
'population', 'total_bedrooms', 'longitude',
'housing_median_age', 'ocean_proximity']
['median_income', 'households', 'longitude']
-> ['total_rooms', 'median_house_value',
'population', 'total_bedrooms', 'latitude',
'housing_median_age', 'ocean_proximity']
['median_income', 'housing_median_age',
'households', 'ocean_proximity'] ->
['total_rooms', 'median_house_value',
'population', 'total_bedrooms', 'latitude',
'longitude']
['median_income', 'housing_median_age',
'latitude', 'longitude'] ->
['median_house_value']
['median_income', 'housing_median_age',
'latitude'] -> ['ocean_proximity']
['median_income', 'housing_median_age',
'longitude'] -> ['ocean_proximity']
['total_rooms', 'median_house_value',
'population'] -> ['median_income',
'total_bedrooms', 'households', 'latitude',
'longitude', 'housing_median_age',
'ocean_proximity']
['total_rooms', 'median_house_value',
'total_bedrooms'] -> ['housing_median_age',
'ocean_proximity']
['total_rooms', 'median_house_value',
'households'] -> ['median_income',
'population', 'total_bedrooms', 'latitude',
'longitude', 'housing_median_age',
'ocean_proximity']
['total_rooms', 'median_house_value',
'latitude'] -> ['housing_median_age',
'ocean_proximity']
['total_rooms', 'median_house_value',
'longitude'] -> ['median_income',
'population', 'total_bedrooms', 'households',
```

```
'latitude', 'housing_median_age',
'ocean_proximity']
['total_rooms', 'population',
'total_bedrooms'] -> ['median_income',
'median_house_value', 'households',
'latitude', 'longitude',
'housing_median_age', 'ocean_proximity']
['total_rooms', 'population', 'households']
-> ['median_income', 'median_house_value',
'total_bedrooms', 'latitude', 'longitude',
'housing_median_age', 'ocean_proximity']
['total_rooms', 'population', 'latitude']
-> ['median_income', 'median_house_value',
'total_bedrooms', 'households', 'longitude',
'housing_median_age', 'ocean_proximity']
['total_rooms', 'population', 'longitude']
-> ['ocean_proximity']
['housing_median_age', 'total_rooms',
'population', 'longitude'] ->
['median_income', 'median_house_value',
'total_bedrooms', 'households', 'latitude']
['housing_median_age', 'total_rooms',
'population', 'ocean_proximity'] ->
['median_income', 'median_house_value',
'total_bedrooms', 'households', 'latitude',
'longitude']
['housing_median_age', 'total_rooms',
'total_bedrooms', 'households'] ->
['median_income', 'median_house_value',
'population', 'latitude', 'longitude',
'ocean_proximity']
['total_rooms', 'total_bedrooms',
'latitude'] -> ['median_income',
'median_house_value', 'population',
'households', 'longitude',
'housing_median_age', 'ocean_proximity']
['total_rooms', 'total_bedrooms',
'longitude'] -> ['median_income',
'median_house_value', 'population',
'households', 'latitude',
'housing_median_age', 'ocean_proximity']
['total_rooms', 'households', 'latitude']
-> ['median_income', 'median_house_value',
'population', 'total_bedrooms', 'longitude',
'housing_median_age', 'ocean_proximity']
['total_rooms', 'households', 'longitude']
-> ['median_income', 'median_house_value',
'population', 'total_bedrooms', 'latitude',
'housing_median_age', 'ocean_proximity']
['median_house_value', 'population',
'total_bedrooms', 'households'] ->
['median_income', 'total_rooms', 'latitude',
'longitude', 'housing_median_age']
['housing_median_age', 'median_house_value',
'population', 'total_bedrooms'] ->
['median_income', 'total_rooms',
'households', 'latitude', 'longitude',
'ocean_proximity']
['ocean_proximity', 'median_house_value',
'population', 'total_bedrooms'] ->
['median_income', 'total_rooms',
'households', 'latitude', 'longitude',
'housing_median_age']
['median_house_value', 'population',
'households', 'latitude'] ->
['median_income', 'total_rooms',
'total_bedrooms', 'longitude',
'housing_median_age']
['housing_median_age', 'median_house_value',
'population', 'households'] ->
['median_income', 'total_rooms',
'total_bedrooms', 'latitude', 'longitude',
'ocean_proximity']
['median_house_value', 'population',
'latitude'] -> ['ocean_proximity']
['median_house_value', 'population',
'longitude'] -> ['median_income',
'total_rooms', 'total_bedrooms',
'households', 'latitude',
'housing_median_age', 'ocean_proximity']
['median_house_value', 'total_bedrooms',
'households', 'latitude'] ->
['median_income', 'total_rooms',
'population', 'longitude',
'housing_median_age']
['median_house_value', 'total_bedrooms',
'households', 'longitude'] ->
['median_income', 'total_rooms',
'population', 'latitude',
'housing_median_age']
['housing_median_age', 'median_house_value',
'total_bedrooms', 'households'] ->
['median_income', 'total_rooms',
'population', 'latitude', 'longitude',
'ocean_proximity']
['housing_median_age', 'median_house_value',
'total_bedrooms', 'latitude'] ->
['ocean_proximity']
['housing_median_age', 'median_house_value',
'total_bedrooms', 'longitude'] ->
['median_income', 'total_rooms',
'population', 'households', 'latitude',
'ocean_proximity']
['median_house_value', 'households',
'latitude'] -> ['ocean_proximity']
['median_house_value', 'households',
'latitude', 'longitude'] -> ['median_income',
'total_rooms', 'population',
'total_bedrooms', 'housing_median_age']
['median_house_value', 'households',
'longitude'] -> ['ocean_proximity']
['housing_median_age', 'median_house_value',
'households', 'longitude'] ->
['median_income', 'total_rooms',
'population', 'total_bedrooms', 'latitude']
['population', 'total_bedrooms',
'households'] -> ['ocean_proximity']
['housing_median_age', 'population',
'total_bedrooms', 'households'] ->
['median_income', 'total_rooms',
'median_house_value', 'latitude',
'longitude']
['population', 'total_bedrooms', 'latitude']
-> ['median_income', 'total_rooms',
'median_house_value', 'households',
'longitude', 'housing_median_age',
'ocean_proximity']
['population', 'total_bedrooms',
'longitude'] -> ['median_income',
'total_rooms', 'median_house_value',
'households', 'latitude',
'housing_median_age', 'ocean_proximity']
['population', 'households', 'latitude'] ->
['ocean_proximity']
['population', 'households', 'latitude',
'longitude'] -> ['median_income',
'total_rooms', 'median_house_value',
'total_bedrooms', 'housing_median_age']
['housing_median_age', 'population',
'households', 'latitude'] ->
['median_income', 'total_rooms',
'median_house_value', 'total_bedrooms',
'longitude']
['population', 'households', 'longitude'] ->
['ocean_proximity']
['housing_median_age', 'population',
'households', 'longitude'] ->
['median_income', 'total_rooms',
'median_house_value', 'total_bedrooms',
'latitude']
['housing_median_age', 'population',
'longitude'] -> ['ocean_proximity']
['total_bedrooms', 'households', 'latitude',
'longitude'] -> ['median_income',
'total_rooms', 'median_house_value',
'population', 'housing_median_age']
['housing_median_age', 'total_bedrooms',
'households', 'latitude'] ->
['ocean_proximity']
['housing_median_age', 'total_bedrooms',
'households', 'longitude'] ->
['ocean_proximity']
['housing_median_age', 'total_bedrooms',
'latitude', 'longitude'] -> ['median_income',
'total_rooms', 'median_house_value',
'population', 'households']
['latitude', 'longitude'] ->
['ocean_proximity']
```

Loan

['Income', 'CITY'] -> ['House_Ownership', 'Married/Single', 'Car_Ownership']
['Income', 'CITY', 'Age'] -> ['Profession', 'Experience', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS']
['Income', 'CITY', 'Profession'] -> ['Experience', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS']
['Income', 'CITY', 'Experience'] -> ['Profession', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS']
['CURRENT_HOUSE_YRS', 'Income', 'CITY'] -> ['Profession', 'Experience', 'CURRENT_JOB_YRS']
['Income', 'Age'] -> ['House_Ownership', 'Married/Single', 'Car_Ownership']
['Income', 'Age', 'Profession'] -> ['CITY', 'STATE', 'Experience', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS']
['Income', 'Age', 'STATE'] -> ['CITY', 'Profession', 'Experience', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS']
['Income', 'Age', 'Experience'] -> ['CITY', 'Profession', 'STATE', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS']
['CURRENT_HOUSE_YRS', 'Income', 'Age'] -> ['CITY', 'Profession', 'STATE', 'Experience', 'CURRENT_JOB_YRS']
['Income', 'Age', 'Risk_Flag'] -> ['CITY', 'Profession', 'STATE', 'Experience', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS']
['Income', 'Profession'] -> ['House_Ownership', 'Married/Single', 'Car_Ownership']
['Income', 'Profession', 'STATE'] -> ['CITY', 'Experience', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS']
['Income', 'Profession', 'Experience'] -> ['CITY', 'STATE', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS']
['CURRENT_HOUSE_YRS', 'Income', 'Profession'] -> ['CITY', 'STATE', 'Experience', 'CURRENT_JOB_YRS']
['Income', 'STATE'] -> ['House_Ownership', 'Car_Ownership']
['Income', 'Married/Single', 'STATE', 'Experience'] -> ['Profession', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS']
['CURRENT_HOUSE_YRS', 'Income', 'STATE'] -> ['CITY', 'Profession', 'Experience', 'CURRENT_JOB_YRS', 'Married/Single']
['Income', 'Married/Single', 'STATE'] -> ['CITY']
['Income', 'Experience'] -> ['House_Ownership']
['CURRENT_HOUSE_YRS', 'Income', 'Experience'] -> ['CITY', 'Profession', 'STATE', 'CURRENT_JOB_YRS', 'Married/Single', 'Car_Ownership']
['Income', 'CURRENT_JOB_YRS'] -> ['CITY', 'Profession', 'STATE', 'Experience', 'CURRENT_HOUSE_YRS', 'House_Ownership', 'Married/Single', 'Car_Ownership']
['CURRENT_HOUSE_YRS', 'Income', 'Car_Ownership'] -> ['House_Ownership']
['CITY'] -> ['STATE']
['CITY', 'Age', 'Profession', 'Experience', 'CURRENT_JOB_YRS'] -> ['House_Ownership']
['CITY', 'Age', 'Profession', 'Experience', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS'] -> ['Car_Ownership']
['CITY', 'Age', 'Profession', 'Experience', 'CURRENT_HOUSE_YRS'] -> ['House_Ownership', 'Married/Single']
['CITY', 'Age', 'Profession', 'Experience', 'CURRENT_HOUSE_YRS', 'Risk_Flag', 'Car_Ownership'] -> ['CURRENT_JOB_YRS']
['CITY', 'Age', 'Profession', 'Experience', 'Risk_Flag', 'Married/Single'] -> ['House_Ownership']
['Age', 'Profession', 'STATE', 'Experience', 'CURRENT_JOB_YRS', 'CURRENT_HOUSE_YRS', 'Risk_Flag', 'Car_Ownership'] -> ['Married/Single']