Semi-supervised Tabular Classification via In-context Learning of Large Language Models

Jaehyun Nam [1] **Woomin Song** [1] **Seong Hyeon Park** [1] **Jihoon Tack** [1]
Sukmin Yun [2] **Jaehyung Kim** [1] **Jinwoo Shin** [1]

Abstract

Learning with limited labeled tabular samples is an important problem for industrial machine learning applications, as acquiring annotations for tabular data is often too costly. On the other hand, recent remarkable progress in natural language processing has evidenced that such an issue can be circumvented by using pre-trained large language models (LLMs). Motivated by this, we ask whether LLMs can help to handle the limited labeled data in the tabular domain as well. As a positive answer, we propose a novel semi-supervised tabular learning framework, coined *Self-generated PROmpts from Unlabeled Tables (SPROUT)*, which utilizes unlabeled data in conjunction with LLMs. Our main idea is to exploit the in-context learning capabilities of LLMs to effectively extract transferable knowledge from unlabeled tabular samples. Specifically, SPROUT generates in-context prompts from unlabeled tables by identifying a column feature that exhibits a strong correlation with the actual target label, thereby creating examples that pertain to the true target tasks. In addition, we demonstrate how a language prior can facilitate knowledge transfer from heterogeneous data sources, enhancing performance of target datasets and mitigating the challenges posed by varying input formats. Experimental results show that SPROUT yields substantial performance improvements over previous methods across various tabular benchmarks.

1. Introduction

Learning with a limited number of labeled samples is often a critical requirement for real-world machine learning applications. While numerous semi-supervised learning approaches have been thoroughly explored in domains such as images (Assran et al., 2021; Pham et al., 2021; Tarvainen & Valpola, 2017) and languages (Chen et al., 2021; Deschacht & Moens, 2009), research on tabular data has only recently begun to gain traction (Nam et al., 2023; Yoon et al., 2020), despite its wide-ranging impact across various industries (Guo et al., 2017; Ulmer et al., 2020; Zhang et al., 2020). Semi-supervised tabular learning is particularly important, because many tabular datasets require substantial annotation efforts, as exemplified by credit risk assessment in financial datasets (Clements et al., 2020), and present difficulties in obtaining new samples for emerging tasks, such as identifying patients with rare or novel diseases (Peplow, 2016) like the initial cases of COVID-19 infection (Zhou et al., 2020).

On the other hand, recent advancements in natural language processing suggest that such an issue can be mitigated by employing pre-trained large language models (LLMs). In particular, LLMs have exhibited their effectiveness even with minimal task-specific instructions in the language domain (Brown et al., 2020; Dong et al., 2022; Wei et al., 2022), suggesting their capacity to address the challenges of limited labeled data. Furthermore, the inherent flexibility of language makes it possible to transform tabular data into language in a natural and direct way. This opens up the possibility of using LLMs for tabular learning, which could lead to a number of benefits. Notably, some recent studies (Dinh et al., 2022; Hegselmann et al., 2023) have investigated the performance of fine-tuned LLMs (Brown et al., 2020; Sanh et al., 2022), reporting competitive results compared to prior tabular learning methods.

Motivated by this, we ask whether unlabeled tabular data and pre-trained large language models can be integrated to offer an innovative solution to the challenge of limited labeled tabular data. Specifically, we investigate the in-context learning (ICL) capabilities of LLMs (Dong et al., 2022), as in-context learning provides a practical advantage by facilitating the rapid prototyping of pre-trained large language models without necessitating fine-tuning (Zhao et al., 2021). As a positive response, we suggest further exploiting the benefits of ICL by prompting examples generated from unlabeled tables, leveraging the knowledge transfer potential inherent in neural networks.

[1]KAIST [2]MBZUAI. Correspondence to: Jaehyun Nam <jaehyun.nam@kaist.ac.kr>, Jinwoo Shin <jinwoos@kaist.ac.kr>.

Work presented at the ES-FoMo Workshop at ICML 2023.

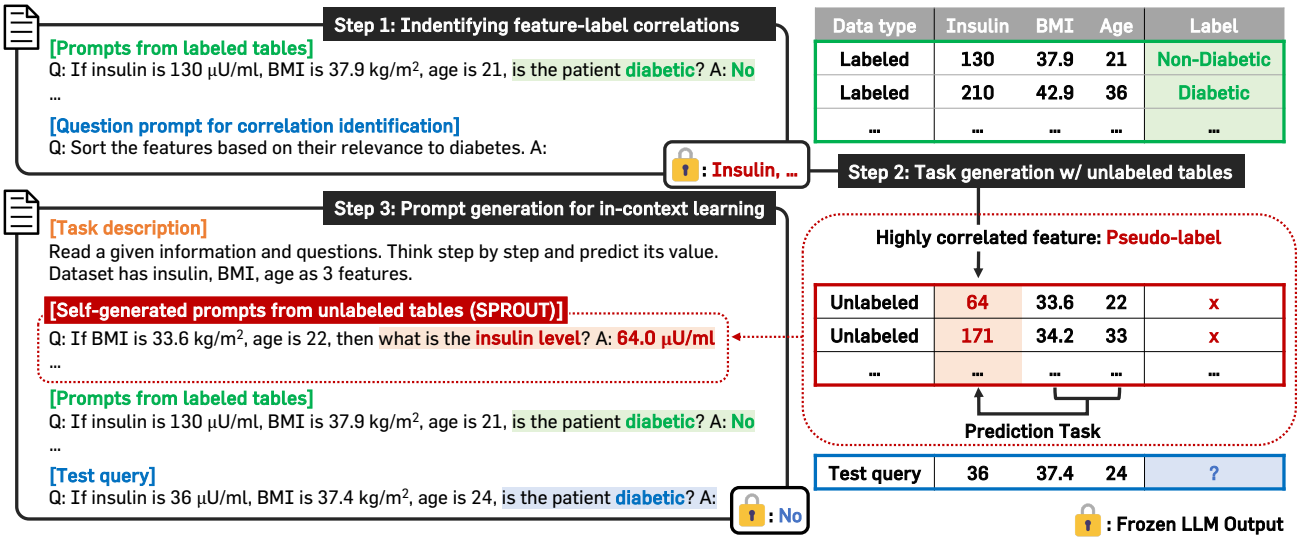

Figure 1. An overview of the proposed *Self-generated PROmpts from Unlabeled Tables (SPROUT).*

Contribution. We propose a novel semi-supervised tabular learning framework, entitled *Self-generated PROmpts from Unlabeled Tables* (SPROUT), as illustrated in Figure 1. Our key idea is to leverage the ICL capabilities of LLMs to effectively extract transferable knowledge from unlabeled datasets. To achieve this, we generate prompts from unlabeled tabular data, by identifying a column feature that exhibits a strong correlation with the actual target label. Then, we create examples that pertain to the true target tasks (*e.g.*, using "tumor size" as a substitute label for "breast cancer type"). Specifically, our method begins by providing an LLM with a few labeled samples and asking it to determine the most important column feature for target prediction. Next, we generate prompts that predict the chosen column feature using the remaining column features, ensuring that the constructed examples are closely related to the actual target task. Finally, we integrate the generated prompts with descriptions of a limited number of labeled samples and apply ICL to enhance learning outcomes.

In addition, certain tabular datasets lack informative column descriptions, limiting the use of LLM's language prior. For example, the credit approval dataset (Asuncion & Newman, 2007) uses obfuscated feature names and values to protect data confidentiality. Hence, it becomes necessary to use generic indicators, such as "input variable", instead of real column names. However, the optimal choice of these indicators is uncertain, while ICL is sensitive to prompt modifications (Min et al., 2022; Wei et al., 2022). To address this challenge, we propose an unsupervised method to discover LLM-friendly, generic feature and target indicators (*e.g.*, "output variable"). Our proposed method minimally alters the prompt, yet exhibits effectiveness.

We verify the effectiveness of the proposed method, by conducting comprehensive evaluations on diverse datasets sourced from Dinh et al. (2022). Our experimental results show that SPROUT significantly and consistently outperforms existing methods, including self-supervised learning (Yoon et al., 2020) and unsupervised meta-learning (Nam et al., 2023) approaches, particularly in few-shot semi-supervised settings (Nam et al., 2023). Furthermore, we also show that SPROUT robustly handles missing values by simply omitting it from the prompt. In addition, SPROUT successfully processes multiple tabular datasets concurrently by leveraging the flexibility of language and language prior within LLMs. In essence, SPROUT enhances the performance of the target dataset by transferring knowledge from heterogeneous data sources.

2. Method

In this section, we propose an effective semi-supervised tabular classification framework that utilizes the in-context learning (ICL) capabilities of large language models (LLMs) to effectively extract transferable knowledge from unlabeled samples. In a nutshell, our framework generates prompts from unlabeled tabular datasets, followed by ICL-based classification using descriptions of labeled samples. We first briefly describe our problem setup, and then the core component, coined *Self-generated PROmpts from Unlabeled Tables (SPROUT)*, which generates effective prompts from unlabeled tabular data for use in ICL (Section 2.1). Additionally, we propose an approach to discover LLM-friendly features and target output indicators (*e.g.*, "input variable"). We find that this enhances the compatibility of tabular prompts with LLMs (Appendix C).

Problem setup. We first describe the problem setup of our interest: semi-supervised tabular classification. A labeled dataset $\mathcal{D}_l = \{(\mathbf{x}_{l,i}, \mathbf{y}_{l,i})\}_{i=1}^{N_l} \subseteq \mathcal{X} \times \mathcal{Y}$ and an unlabeled dataset $\mathcal{D}_u = \{\mathbf{x}_{u,i}\}_{i=1}^{N_u} \subseteq \mathcal{X}$ with column name set $F = \{f_1, \cdots, f_d, f_{d+1}\}$ are given, where $\mathbf{x}_{l,i}$, $\mathbf{x}_{u,i}$ are d-dimensional feature vectors, which correspond to the value in the respective table columns (*e.g.*, "Male" and "36.0" are feature values of the "sex" column and "BMI" column, respectively). F stands for the column names in the dataset, including the output feature (*i.e.*, f_{d+1}). These could be natural language descriptions such as "age" or "education," or, in the absence of column name descriptions, they could be generic indicators like "input variable1," "input variable2," and so forth. Further, $\mathcal{X}$ and $\mathcal{Y}$ represent the tabular input space and the label space. Labels $\mathbf{y} \in \mathcal{Y}$ are provided in the form of natural language annotations (*e.g.*, in Figure 1, "Non-Diabetic" and "Diabetic" are labels of the Diabetes dataset). Here, the cardinality of the labeled dataset is assumed to be much smaller than the unlabeled dataset.

2.1. Semi-supervised in-context learning with SPROUT

We now present SPROUT, a novel approach to improve tabular classification performance by creating an effective prompt generator that encapsulates the context from both labeled and unlabeled data. Based on the information derived from labeled tables, SPROUT employs unlabeled tabular data to generate in-context prompts. This is accomplished by identifying a table column feature that shows a significant correlation with the actual target label, thereby generating examples that are more directly aligned with the true target tasks. This context-rich prompt is then used as the input for the *frozen* language model classifier, `LLM`. Formally, we aim to create a prompt generator function $g : \mathcal{X} \to \mathcal{S}$, with $\mathcal{S}$ denoting the text space, and implement a language model classifier $\texttt{LLM} : \mathcal{S} \to \mathcal{Y}$. Our primary objective is to precisely predict the label $\mathbf{y}_t$ of a test sample $\mathbf{x}_t$ via the function $\texttt{LLM}(g(F, \mathcal{D}_l, \mathcal{D}_u, \mathbf{x}_t))$.

One key aspect that differentiates our approach is that our prompt generator g incorporates the context from the labeled, unlabeled, and test data when generating prompts for the `LLM` classifier. This is contrary to prior ICL methods on the tabular domain (*e.g.*, LIFT-ICL (Dinh et al., 2022)), which rely solely on $\mathcal{D}_l$ to predict the output of $\mathbf{x}_t$, i.e., $\texttt{LLM}(g(F, \mathcal{D}_l, \mathbf{x}_t))$. Namely, our proposed method utilizes $\mathcal{D}_u$, the unlabeled data, in the prompt generator for ICL, leveraging the inherent knowledge from the unlabeled data, which significantly enhances the effectiveness of ICL.

To begin, we first describe the process of tabular data serialization, a method that maps the tabular data into a text space. Following this, we describe our primary, carefully designed algorithm. We start by identifying the specific feature with the highest correlation to the actual target label. Utilizing this feature, we then generate prompts. These prompts are comprised of information derived from unlabeled data, which is subsequently formatted according to the task via the serialization technique.

Tabular serialization. In order to transform the tabular data to natural language, following previous literature (Dinh et al., 2022; Hegselmann et al., 2023), we formally define the serialization function $\texttt{serialize} : \mathcal{X} \to \mathcal{S}$ as follows:

$$\begin{aligned}&\texttt{serialize}(\mathbf{x}, \mathbf{y}, \{f_i\}_{i=1}^{d}, f_{d+1}) = \\ &\text{"Q : When } f_1 \text{ is } x_1,\ f_2 \text{ is } x_2, \cdots,\ f_d \text{ is } x_d, \\ &\ \text{then what is } f_{d+1}\text{? A : } \mathbf{y}\text{."}\end{aligned}$$

The result is a serialized form of the table to natural language, where it presents the data as a question.

Correlation identification. Our method starts by identifying the feature f that holds the highest correlation with the target variable, denoted as $\mathbf{y}$. This correlation is measured among all the features $\{f_j\}_{j=1}^{d}$. This identification is carried out by supplying the input, $\texttt{merge}(\{\texttt{serialize}(\mathbf{x}_{l,i}, \mathbf{y}_{l,i}, \{f_j\}_{j=1}^{d}, f_{d+1})\}_{i \in \mathcal{D}_l})$, to the `LLM` classifier. Specifically, the input is structured with the `question prompt`, "Sort f_i that are more related to the f_{d+1}." Here, the function `merge` collates the inputs, arranging them row by row within a single prompt, incorporating multiple sentences within a single prompt. The selected column feature f_k is then computed by:

$$\begin{aligned}f_k = &\texttt{LLM}\big(\texttt{merge}\big(\\ &\{\texttt{serialize}(\mathbf{x}_{l,i}, \mathbf{y}_{l,i}, \{f_j\}_{j=1}^{d}, f_{d+1})\}_{i \in \mathcal{D}_l} \cup \\ &[\text{question prompt}]\big)\big).\end{aligned}$$

Prompt generation from unlabeled tables. Our main idea is to generate prompts from unlabeled data based on a highly correlated feature f_k which we use as a pseudo-label. The rationale behind using f_k, as a pseudo-label stems from the intuition that the most correlated feature likely can form tasks that closely resemble the original classification task. For instance, predicting "Diabetes" from "BMI" and "Age" is similar to predicting "Insulin" using the same features (Nam et al., 2023). Thus, we generate prompts that predict the value of f_k from remaining features, which we refer to as `SPROUT` prompts.

However, utilizing all unlabeled data can be difficult as LLMs may limit the input prompt size. Thus, we simply consider only a subset of the unlabeled data by selecting a small, fixed number of samples that are closest to each labeled sample in terms of Euclidean distance.[1] We denote by $\mathcal{I}_u \subset [1, N_u]$ to represent the indices of the selected

[1]Data is vectorized by one-hot encoding categorical features and applying min-max scaling to all features.

Table 1. Few-shot test accuracy (%) on 9 datasets from the OpenML repository (Vanschoren et al., 2014). # shot indicates the number of labeled samples per class. For the baselines, we report the average test accuracy over 100 different seeds. We report the average accuracy and standard deviation over 5 different seeds for our method due to the high cost of OpenAI API. The bold denotes the highest average score. † denotes experiments done with GPT-4 due to the prompt size limit of ChatGPT.

Method	Breast	TAE	Vehicle	Hamster	Customers	LED	Pollution	Diabetes	Car	Avg.
# shot = 1										
CatBoost (Prokhorenkova et al., 2018)	57.64	34.29	37.60	51.87	64.12	49.71	63.58	58.60	32.33	49.97
LR	61.23	37.35	36.11	51.07	61.34	54.70	63.67	57.61	36.95	51.11
kNN	61.88	37.26	36.22	51.00	63.81	51.49	63.67	58.56	31.51	50.60
VIME+LR (Yoon et al., 2020)	57.38	37.87	35.32	51.53	62.48	52.99	63.33	56.95	34.51	50.26
VIME+kNN (Yoon et al., 2020)	57.38	38.16	34.46	51.53	62.47	53.30	63.33	58.35	33.38	50.26
STUNT (Nam et al., 2023)	53.04	36.87	34.58	51.73	65.14	48.55	63.00	61.08	36.48	50.05
LIFT-ICL (Dinh et al., 2022)	66.43	30.97	37.18	48.00	60.91	20.60†	58.33	62.60	69.13	50.46
SPROUT	**68.93**$_{\pm 6.13}$	**43.23**$_{\pm 7.07}$	**39.88**$_{\pm 2.51}$	**58.67**$_{\pm 5.58}$	**87.27**$_{\pm 3.69}$	**55.40**†$_{\pm 6.15}$	**65.00**$_{\pm 3.73}$	**68.44**$_{\pm 5.02}$	**71.40**$_{\pm 1.79}$	**62.02**
# shot = 5										
CatBoost (Prokhorenkova et al., 2018)	57.63	39.71	**51.94**	56.33	81.40	67.04	70.58	64.94	46.96	59.61
LR	61.21	43.42	46.52	51.60	60.82	70.10	73.33	64.19	53.29	58.28
kNN	62.33	44.65	43.47	54.53	64.92	71.17	72.83	67.32	49.62	58.98
VIME+LR (Yoon et al., 2020)	60.89	42.84	47.34	52.80	66.07	68.30	75.50	64.29	52.37	58.93
VIME+kNN (Yoon et al., 2020)	64.12	41.68	40.35	53.47	63.42	**71.59**	70.33	66.94	49.74	57.96
STUNT (Nam et al., 2023)	61.30	40.77	40.46	52.87	66.44	66.97	70.92	69.88	51.73	57.93
LIFT-ICL (Dinh et al., 2022)	67.86	35.48	39.18†	58.67	88.18†	54.60†	65.00	69.20	70.81	61.00
SPROUT	**72.85**$_{\pm 1.96}$	**45.81**$_{\pm 1.44}$	41.41†$_{\pm 4.85}$	**64.00**$_{\pm 7.60}$	**89.55**†$_{\pm 0.85}$	64.00†$_{\pm 2.00}$	**76.67**$_{\pm 3.73}$	**71.44**$_{\pm 2.26}$	**72.08**$_{\pm 1.03}$	**66.42**

unlabeled data, which we incorporate into the task prompt generation. Formally, we define the prompt generation process SPROUT as:

$$\texttt{SPROUT}\big(\mathbf{x}_{u,i}, \{f_j\}_{j=1}^{d}, k\big) =$$
"Q : When f_1 is $x_1, \cdots, f_{k-1}$ is x_{k-1}, f_{k+1} is $x_{k+1}, \cdots,$ f_d is x_d, then what is f_k? A : x_k."

Finally, we propose our prompt generator g, which incorporates a prompt that describes the task, the SPROUT derived from unlabeled data, and the information from the labeled samples. Additionally, it includes a test query prompt which is "When f_1 is $x_{t,1}$, f_2 is $x_{t,2}, \cdots, f_d$ is $x_{t,d}$, then what is f_{d+1}?," *i.e.*, the conventional test query prompt for ICL. Formally, we define our prompt generator g as:

$$\begin{aligned} g(F, \mathcal{D}_l, \mathcal{D}_u, \mathbf{x}_t) = \texttt{merge}\big(&[\texttt{task description}] \\ &\cup \big\{\texttt{SPROUT}\big(\mathbf{x}_{u,i}, \{f_j\}_{j=1}^{d}, k\big)\big\}_{i \in \mathcal{I}_u} \\ &\cup \big\{\texttt{serialize}\big(\mathbf{x}_{l,i}, \mathbf{y}_{l,i}, \{f_j\}_{j=1}^{d}, f_{d+1}\big)\big\}_{i \in \mathcal{D}_l} \\ &\cup [\text{test query}]\big). \end{aligned}$$

In-context learning with SPROUT. After generating a prompt using g, we put the generated prompt into the LLM to predict the label $\mathbf{y}_{\text{pred}}$ of the test query $\mathbf{x}_t$. Formally, $\mathbf{y}_{\text{pred}} = \texttt{LLM}\big(g(F, \mathcal{D}_l, \mathcal{D}_u, \mathbf{x}_t)\big)$.

3. Experiments

In this section, we validate the effectiveness of our proposed method in various semi-supervised learning scenarios, utilizing diverse tabular datasets sourced from the OpenML repository (Vanschoren et al., 2014). These selected datasets have been previously employed in the in-context learning (ICL) experiments by Dinh et al. (2022). The experimental results reveal that SPROUT consistently outperforms other baseline methods. Next, we conduct ablation studies to verify the impact of each core component of SPROUT (Appendix D). Finally, to further demonstrate the practical applicability of our method, we expand our evaluation to introduce its intriguing properties in two practical scenarios: handling missing values in tabular data and taking benefit from the heterogeneous data sources (Appendix E).

Semi-supervised tabular classification. In this section, we demonstrate the efficacy of SPROUT for semi-supervised classification tasks. Due to the constraints posed by the limited input prompt size of LLMs, we have decided to evaluate our method in the context of few-shot semi-supervised classification, as described in Nam et al. (2023). Our performance evaluation is based on scenarios with one and five labeled samples available per class. As demonstrated in Table 1, SPROUT consistently improves the few-shot semi-supervised tabular classification performance. Note that this improvement is achieved without model updates. To provide a specific example, SPROUT significantly outperforms LR in 1-shot classification, raising the average performance from 51.11% to 62.02%. Additionally, SPROUT consistently achieves superior results, yielding the highest score in all 9 datasets in the 1-shot classification problem, and in 7 out of the 9 datasets in the 5-shot scenario. These results represent an improvement of approximately 10.9% and 5.4% over the best performing baselines, respectively. The success of SPROUT is attributed to its effective use of the ICL capabilities of LLMs. By constructing effective prompts from unlabeled set, SPROUT is able to extract useful knowledge from the unlabeled set in an ICL manner.

References

Assran, M., Caron, M., Misra, I., Bojanowski, P., Joulin, A., Ballas, N., and Rabbat, M. Semi-supervised learning of visual features by non-parametrically predicting view assignments with support samples. In *IEEE International Conference on Computer Vision*, 2021.

Asuncion, A. and Newman, D. UCI machine learning repository, 2007.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 2020.

Chen, L., Garcia, F., Kumar, V., Xie, H., and Lu, J. Industry scale semi-supervised learning for natural language understanding. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2021.

Clements, J. M., Xu, D., Yousefi, N., and Efimov, D. Sequential deep learning for credit risk monitoring with tabular financial data. *arXiv preprint arXiv:2012.15330*, 2020.

Deschacht, K. and Moens, M.-F. Semi-supervised semantic role labeling using the latent words language model. In *EMNLP*, 2009.

Dinh, T., Zeng, Y., Zhang, R., Lin, Z., Gira, M., Rajput, S., Sohn, J.-y., Papailiopoulos, D., and Lee, K. Lift: Language-interfaced fine-tuning for non-language machine learning tasks. *Advances in Neural Information Processing Systems*, 2022.

Do, C. B. and Ng, A. Y. Transfer learning for text classification. *Advances in neural information processing systems*, 2005.

Dong, Q., Li, L., Dai, D., Zheng, C., Wu, Z., Chang, B., Sun, X., Xu, J., and Sui, Z. A survey for in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

Guo, H., Tang, R., Ye, Y., Li, Z., and He, X. Deepfm: A factorization-machine based neural network for ctr prediction. In *International Joint Conferences on Artificial Intelligence*, 2017.

Hegselmann, S., Buendia, A., Lang, H., Agrawal, M., Jiang, X., and Sontag, D. Tabllm: few-shot classification of tabular data with large language models. In *International Conference on Artificial Intelligence and Statistics*, 2023.

Min, S., Lyu, X., Holtzman, A., Artetxe, M., Lewis, M., Hajishirzi, H., and Zettlemoyer, L. Rethinking the role of demonstrations: What makes in-context learning work? In *EMNLP*, 2022.

Nam, J., Tack, J., Lee, K., Lee, H., and Shin, J. STUNT: Few-shot tabular learning with self-generated tasks from unlabeled tables. In *International Conference on Learning Representations*, 2023.

OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Peplow, M. The 100,000 genomes project. *British Medical Journal*, 2016.

Pham, H., Dai, Z., Xie, Q., Luong, M.-T., and Le, Q. V. Meta pseudo labels. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2021.

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., and Gulin, A. Catboost: unbiased boosting with categorical features. In *Advances in Neural Information Processing Systems*, 2018.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. Language models are unsupervised multitask learners. *OpenAI blog*, 2019.

Raina, R., Ng, A. Y., and Koller, D. Constructing informative priors using transfer learning. In *Proceedings of the 23rd international conference on Machine learning*, 2006.

Sanh, V., Webson, A., Raffel, C., Bach, S. H., Sutawika, L., Alyafeai, Z., Chaffin, A., Stiegler, A., Scao, T. L., Raja, A., et al. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations*, 2022.

Tarvainen, A. and Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, 2017.

Ucar, T., Hajiramezanali, E., and Edwards, L. Subtab: Subsetting features of tabular data for self-supervised representation learning. In *Advances in Neural Information Processing Systems*, 2021.

Ulmer, D., Meijerink, L., and Cinà, G. Trust issues: Uncertainty estimation does not enable reliable ood detection on medical tabular data. In *Machine Learning for Health*, 2020.

Vanschoren, J., van Rijn, J. N., Bischl, B., and Torgo, L. OpenML. *ACM SIGKDD Explorations Newsletter*, 2014.

Verma, V., Kawaguchi, K., Lamb, A., Kannala, J., Solin, A., Bengio, Y., and Lopez-Paz, D. Interpolation consistency training for semi-supervised learning. *Neural Networks*, 2022.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E., Le, Q., and Zhou, D. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, 2022.

Yi, J., Lee, J., Kim, K. J., Hwang, S. J., and Yang, E. Why not to use zero imputation? correcting sparsity bias in training neural networks. In *International Conference on Learning Representations*, 2020.

Yoon, J., Jordon, J., and Schaar, M. Gain: Missing data imputation using generative adversarial nets. In *International conference on machine learning*, 2018.

Yoon, J., Zhang, Y., Jordon, J., and van der Schaar, M. Vime: Extending the success of self- and semi-supervised learning to tabular domain. In *Advances in Neural Information Processing Systems*, 2020.

Zhang, S., Yao, L., Sun, A., and Tay, Y. Deep learning based recommender system. *Association for Computing Machinery (ACM) Computing Surveys*, 2020.

Zhao, Z., Wallace, E., Feng, S., Klein, D., and Singh, S. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, 2021.

Zhou, P., Yang, X.-L., Wang, X.-G., Hu, B., Zhang, L., Zhang, W., Si, H.-R., Zhu, Y., Li, B., Huang, C.-L., Chen, H.-D., Chen, J., Luo, Y., Guo, H., Jiang, R.-D., Liu, M.-Q., Chen, Y., Shen, X.-R., Wang, X., Zheng, X.-S., Zhao, K., Chen, Q.-J., Deng, F., Liu, L.-L., Yan, B., Zhan, F.-X., Wang, Y.-Y., Xiao, G.-F., and Shi, Z.-L. A pneumonia outbreak associated with a new coronavirus of probable bat origin. *Nature*, 2020.

A. Related work

Utilizing unlabeled tabular data. Researchers have developed a number of methods for training generalizable representations of tabular datasets using unlabeled data. Pioneering this field, Yoon et al. (2020) targeted self-supervised learning on tabular datasets, introducing an approach that involves corrupting random features and predicting the corrupted locations in terms of both rows and columns. Moreover, Ucar et al. (2021) demonstrated that implementing an effective combination of three pretext task losses (*i.e.*, reconstruction loss, contrastive loss, and distance loss) could yield state-of-the-art performance. Furthermore, Nam et al. (2023) proposed an unsupervised meta-learning framework to tackle few-shot semi-supervised learning problems. Compared to prior works, we propose a semi-supervised framework for tabular data using large language models (LLMs), which applies in-context knowledge transfer, enhancing the effective exploitation of unlabeled tables.

Tabular learning with large language models. Recent advances in LLMs have provided an impetus to explore their potential for tabular learning. Dinh et al. (2022) investigated the performance of fine-tuned GPT-3 models (Brown et al., 2020) on tabular data. Extending this line of research, Hegselmann et al. (2023) conducted a comprehensive analysis using the T0 model (Sanh et al., 2022), leveraging the language prior in LLMs. Their analysis extended to sample efficiency considerations, even conducting zero-shot experiments. Inspired by these preceding studies, our work proposes a method for the effective exploitation of unlabeled data - an aspect overlooked in prior research. By integrating the utilization of unlabeled data with LLMs, we aim to enhance performance in semi-supervised learning scenarios significantly.

In-context learning. As model and dataset sizes increase (Brown et al., 2020; Radford et al., 2019), LLMs have exhibited the capability for in-context learning (ICL), where they draw knowledge from a handful of contextual examples. For example, Wei et al. (2022) have illustrated the competency of LLMs in solving mathematical reasoning problems via ICL. The ICL process begins by employing a small number of examples to establish a contextual framework, typically constructed using natural language templates. Following this, a query question and a contextual demonstration are combined to form a prompt, which is subsequently fed to the LLMs for prediction. Notably, ICL does not necessitate parameter updates and directly carries out predictions using LLMs, enabling easy implementation for large-scale real-world tasks. In our work, we delve deeper into the potential of ICL by examining its performance on semi-supervised tabular classification tasks, using unlabeled data as a source for creating effective demonstrations.

B. Common setup and baselines

For all datasets, we use 80% of the data for training, which is all unlabeled except for a limited labeled samples, while the other 20% is used for testing. Following Ucar et al. (2021), categorical features are one-hot encoded for the baselines, followed by min-max scaling. To verify our method, we compare the performance with the competitive and effective baselines subsampled from Nam et al. (2023). In particular, we consider supervised learning baselines including CatBoost (Prokhorenkova et al., 2018), Logistic Regression (LR), and the nearest neighbor classifier operating on the prototype of the input data (kNN), that do not utilize unlabeled data. The semi-supervised learning baseline VIME (Yoon et al., 2020) is also considered where the model is initially pre-trained and then evaluated using labeled samples with LR and kNN. We deliberately exclude methods requiring careful hyperparameter tuning, due to their inherent sensitivity to hyperparameters (*e.g.*, MPL (Pham et al., 2021), MT (Tarvainen & Valpola, 2017), ICT (Verma et al., 2022)) and over-fitting issues. Our problem setup, characterized by limited labeled data, does not lend itself well to hyperparameter tuning in real-world scenarios, due to the absence of labeled validation set. We also consider STUNT (Nam et al., 2023), the state-of-the-art few-shot semi-supervised tabular method. Finally, we consider LIFT (Dinh et al., 2022) in ICL setting as a representative method for leveraging the power of large language models (LLMs). For all experiments using LLMs, we use the GPT models provided by the OpenAI API. Specifically, `text-davinci-003` is used to identify correlation in SPROUT, and ChatGPT (*i.e.*, `gpt-3.5-turbo`) is used for all experiments, unless stated otherwise.

C. Generic indicator discovery with SPROUT

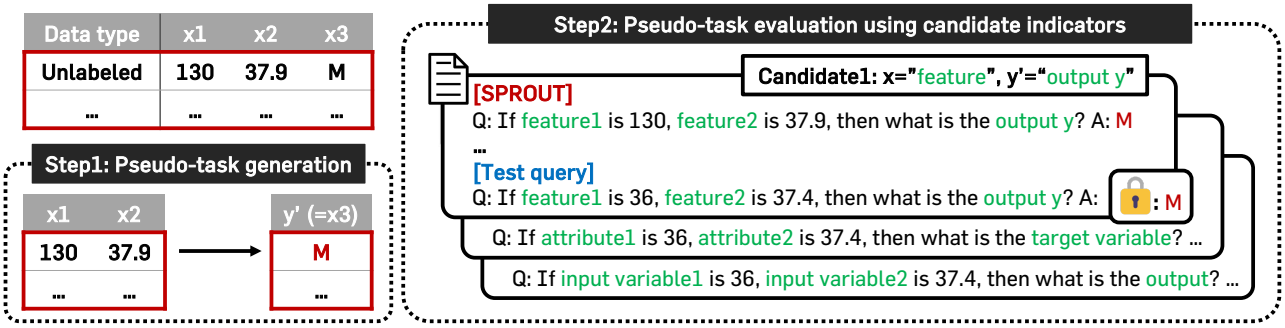

Figure 2. An overview of the proposed generic indicator discovery with SPROUT.

Indeed, LLMs require explicit column descriptions to effectively exploit the language prior, but informative descriptions are often absent in tabular datasets (Asuncion & Newman, 2007). Consequently, researchers are forced to use the *generic indicator*, a prompt that is used to substitute (or pretend as) actual column names, for instance, "independent variable" (Dinh et al., 2022). However, in the context of LLMs (Min et al., 2022; Wei et al., 2022), determining the most suitable generic indicator remains uncertain, while ICL is sensitive to prompt modifications (Wei et al., 2022). To tackle this challenge, we propose an unsupervised method for discovering generic indicators that are more compatible with LLMs. In a nutshell, we generate indicator pairs that indicate the features of input $\mathbf{x}$ and $\mathbf{y}$, respectively. Then, we evaluate each indicator pair through the pseudo-task generated by SPROUT (see Figure 2).

As a detail, in our proposed method, we first ask the LLM to generate candidates of generic indicators $F' = \{f'_1, f'_2, \cdots, f'_d, f'_{d+1}\}$, where $\{f'_j\}_{j=1}^{d}$ are for input features (*e.g.*, "input variable1", "independent variable1") and f'_{d+1} is for the target value (*e.g.*, "output variable", "dependent variable"). We then sample a subset of the unlabeled set $\mathbf{x}'_u \in \mathcal{D}'_u \subset \mathcal{D}_u$ and generate a pseudo-label $\mathbf{y}'_u$. If $\mathbf{x}'_u$ includes categorical features, we select the column feature as $\mathbf{y}'_u$ that has both (i) the smallest number of categories and (ii) the smallest variance between categories. Given a task $\{\mathbf{x}'_{u,i}, \mathbf{y}'_{u,i}\}_i$, we sample two disjoint sets, $\mathcal{S}$ and $\mathcal{Q}$. We then use ICL with set $\mathcal{S}$ as labeled samples to predict the pseudo-label of set $\mathcal{Q}$ (*i.e.*, test query set), using the `SPROUT` function. For convenience, we denote the selected categorical column index as k'. Then formally,

$$\mathbf{y}'_{\texttt{pred}} = \texttt{LLM}\big(\texttt{merge}\big(\{\texttt{SPROUT}(\mathbf{x}'_{u,i}, \{f'_j\}_{j=1}^{d}, k'\}_{i\in\mathcal{S}} \cup [\text{test query}]\big)\big) \text{ where } f'_{k'} = f'_{d+1}.$$

Among the generated F', the candidate that most accurately predicts the pseudo-label of test queries in $\mathcal{Q}$ is then selected. In the case where $\mathbf{x}'_u$ consists only of numerical features, we employ STUNT (Nam et al., 2023) to generate $\mathbf{y}'_u$ from the unlabeled dataset. The overall process remains the same. Discovering generic indicators with our proposed method actually improves performance (see Table 2c for results) because we discover indicators that are compatible with the features in the table in terms of LLMs.

D. Ablation studies

Table 2. The effect of in-context learning (ICL) with labeled and unlabeled data in SPROUT, and the effect of formatting methods for the unlabeled data. We compare 1-shot test accuracy (%) on three datasets from the OpenML repository (Vanschoren et al., 2014). We report the average accuracy over 5 different seeds.

(a) Effect of ICL with labeled & unlabeled data.

Method	Breast	TAE	Customers
LR	61.23	37.35	61.34
Ours (generic)			
+ labeled	55.00	35.48	62.73
+ unlabeled	**58.22**	**41.94**	**70.23**
Ours (descriptive)			
zero-shot	60.71	29.03	43.18
+ labeled	67.50	34.19	65.14
+ unlabeled	**68.93**	**43.23**	**87.27**

(b) Formatting methods for unlabeled data.

Method	Breast	TAE	Customers
w/o unlabeled	55.00	35.48	62.73
Plain	55.36	40.00	68.86
Random Target	53.57	38.07	69.77
Identified Target	**58.22**	**41.94**	**70.23**

(c) Effect of discovered indicator.

Method	Breast	TAE	Customers
Ours (base model)	53.21	**35.48**	61.36
+ discovered indicator	**55.00**	**35.48**	**62.73**

In this section, we analyze the components of SPROUT on the three OpenML datasets (Vanschoren et al., 2014): Breast, TAE, and Customers. We mainly focus on: (i) verifying the efficacy of in-context learning with unlabeled data, (ii) assessing the effectiveness of SPROUT in discovering generic indicators, and (iii) the effect of LLM-friendly informative real column name descriptions.

Effectiveness of in-context learning with unlabeled data. We first investigate the effect of in-context learning with the unlabeled data with SPROUT in Table 2a. As an effort to isolate the impact of the language descriptions in the column names and the categorical data with texts, we present the results of the generic model ("Ours (generic)" in Table 2a) where the language descriptions are removed. For instance, we employ generic indicators such as "input variables" and "output variables" for the column names and substitute all categorical values with random alphabetical symbols (*e.g.*, feature "Summer" is replaced with "A" in the TAE dataset). As shown in Table 2a, leveraging unlabeled data via in-context learning significantly improves performance for all the cases with and without the language descriptions (*e.g.*, 65.14%→87.27% and 62.73%→70.23% in Customers, respectively).

We also verify the efficacy of our correlation identification method in Table 2b, which seeks the most correlated feature to the target while generating the task prompt. For comparison, we consider a basic, naïve method ("Plain" in Table 2b) that serializes the unlabeled data without Q-A pairs, *e.g.*, "f1 is x1, f2 is x2, $\cdots$, fd is xd.", and the model that randomly selects the target ("Random Target" in Table 2b). Our results show that serializing the unlabeled data with the identified target boosts the in-context learning performance by the largest amount (*e.g.*, 35.48%→41.94% in TAE, 62.73%→70.23% in Customers datasets), while other methods yield sub-optimal improvements. We emphasize that our careful task prompt construction from unlabeled tabular data closely mimics the actual target task, thereby offering a considerable advantage in predicting the test query.

Effectiveness of discovered generic indicators. Table 2c ablates our generic indicator discovery method, particularly useful in the instances where column descriptions are not informative. For instance, our method reveals that for the Breast dataset, the generic indicators $F = \{\text{input variable1}, \cdots, \text{input variabled}\}$ and $f_{d+1} =$ "output variable" are more beneficial than $F = \{\text{x1}, \cdots, \text{xd}\}$ and $f_{d+1} =$ "y value". By simply replacing these indicators, we observe a consistent improvement, for example, an increase from 53.21% to 55.00% on the Breast dataset.

Effectiveness of the language descriptions in large language models. Another interesting trend observed in Table 2a is a significant enhancement in the accuracy due to the language descriptions, as exemplified by a rise from 70.23%→87.27% on the Customers dataset. To leverage the language prior of LLMs, we recommend practitioners to thoughtfully employ the detailed column descriptions, if provided. For instance, on the Customers dataset, one can serialize tabular data such as "feature1 is 3191.0" into a more descriptive form like "annual spending on fresh product is 3191.0". Intriguingly, even in the absence of the language descriptions—hence relying solely on a generic prompt that utilizes only basic symbols for categorical data and generic column indicators—SPROUT consistently outperforms or at least achieves competitive performance compared to the baselines, *e.g.*, 70.23% on the Customers dataset, which surpasses performances of all the baselines.

E. Handling various practical scenarios with SPROUT

Table 3. Classification performances in the missing value scenario. We report the 1-shot test accuracy (%) on the two datasets from the OpenML repository (Vanschoren et al., 2014). For each dataset, we randomly remove 50% of the feature values. We simulate the missing values with 5 different seeds and report the average test accuracy. For the baselines, we apply three imputation methods (zero, mean, median) and report the best results. Bold indicates the performance with the lowest performance drop.

Method	Breast		Diabetes	
	Complete	50% Missing	Complete	50% Missing
CatBoost (Prokhorenkova et al., 2018)	57.64	54.50 (-3.14%)	58.60	54.18 (-4.42%)
LR	61.23	55.84 (-5.39%)	57.61	55.43 (-2.18%)
kNN	61.88	56.80 (-5.08%)	58.56	55.43 (-3.13%)
Ours	**68.93**$\pm$6.13	**67.71**$\pm$6.58 **(-1.22%)**	**68.44**$\pm$5.02	**67.01**$\pm$4.53 **(-1.43%)**

In this section, we introduce extended applications of SPROUT within two practical tabular learning scenarios: specifically, managing missing values (a common challenge in the tabular domain), and knowledge transfer from heterogeneous data sources, utilizing the language priors of LLMs. We highlight the intriguing properties of SPROUT, which is distinguished from the baselines.

Robustness to missing values. In practice, tabular data often contains missing values for various reasons. For instance, biopsy results may not be collected for all patients due to the risks and complications involved in the data collection process (Yoon et al., 2018). Conventionally, missing values are managed using imputation algorithms (Yi et al., 2020; Yoon et al., 2018) in the tabular domain, which estimate missing values from other existing information. The performance of standard tabular machine learning methodologies largely depends on imputation algorithms, as incorrectly estimated data could introduce severe noise.

In contrast, SPROUT naturally handles missing values by simply excluding these values from the input prompt. For example, if the "Age" feature is missing, SPROUT serializes the table to prompt like "`Insulin is 130, BMI is 37.9`." To verify the robustness of SPROUT to missing values in tables, we simulate a scenario where 50% of features are randomly omitted. For the unlabeled data selection process in SPROUT, we employ zero imputation when calculating the distance between labeled and unlabeled samples. As shown in Table 3, SPROUT shows not only superior classification accuracy but also exhibits the smallest performance drop (-1.22% and -1.43% for Breast and Diabetes, respectively). These results highlight that SPROUT is robust to missing values since these values are not required during inference, therefore not being severely affected by incorrect estimations.

Table 4. Classification performances in the transfer scenario from heterogeneous data sources. We provide a 1-shot training set for a target dataset and benchmark the effect of incrementing the additional training samples from a source dataset, from $N = 0$ (no heterogeneous sample) to $N = 10$ (many heterogeneous samples). We report the average test accuracy over 3 different seeds for all methods. Experiments for non-LM baselines (†) are implemented by extending columns for the heterogeneous data with zero-padded values. The bold denotes the highest average score.

Target	Source	Method	N=0	N=2	N=4	N=6	N=8	N=10
Adult	Credit Risk	CatBoost† (Prokhorenkova et al., 2018)	56.00	54.67	60.00	61.33	51.33	49.33
		LR†	54.00	69.33	69.33	66.00	61.33	55.33
		kNN†	54.00	72.00	72.00	57.33	57.33	57.33
		LIFT-ICL (Dinh et al., 2022)	69.33	25.33	35.33	52.00	60.00	43.33
		Ours	**74.67**$\pm$1.89	**75.33**$\pm$1.89	**76.00**$\pm$0.00	**77.33**$\pm$2.49	**79.33**$\pm$3.77	**80.00**$\pm$1.63
	Electricity	CatBoost† (Prokhorenkova et al., 2018)	56.00	50.00	50.67	48.67	45.33	58.00
		LR†	54.00	54.67	50.67	50.00	45.33	58.67
		kNN†	54.00	42.67	42.67	37.33	37.33	42.67
		LIFT-ICL (Dinh et al., 2022)	69.33	60.67	64.67	63.33	58.67	54.00
		Ours	**74.67**$\pm$1.89	**80.00**$\pm$2.83	**76.00**$\pm$2.83	**78.67**$\pm$2.49	**80.00**$\pm$1.63	**81.33**$\pm$2.49
Credit-g	Credit Approval	CatBoost† (Prokhorenkova et al., 2018)	**55.33**	46.67	41.33	46.67	40.67	44.00
		LR†	52.67	49.33	48.00	33.33	42.00	40.00
		kNN†	52.67	**58.67**	41.33	41.33	41.33	24.00
		LIFT-ICL (Dinh et al., 2022)	42.67	49.17	48.17	45.83	46.00	48.67
		Ours	55.00$\pm$4.30	54.50$\pm$2.55	**58.67**$\pm$3.30	**59.33**$\pm$3.30	**59.33**$\pm$2.05	**60.67**$\pm$1.65

Transferring from heterogeneous data sources. We next demonstrate the effect of introducing training samples from heterogeneous data sources. Alongside the semi-supervised method exploiting the patterns inferred from unlabeled data in the homogeneous source (*i.e.*, the same distribution as the test set), transferring knowledge from different sources, referred to as transfer learning (Do & Ng, 2005; Raina et al., 2006), is another reasonable approach to dealing with limited labeled data in practice.

However, merging distinct column sets from diverse sources in the tabular domain demands a heuristic process to create a unified feature set. Such an approach may not generalize well, and require sophisticated designs for different data combinations. In this regard, we find our tabular serialization discussed in Section 2.1 to be a simple and effective method for combining columns from various heterogeneous sources. As tabular data is transformed into natural language, the language model can automatically understand the relations between different features from their descriptions.

To investigate the effect of incorporating heterogeneous data, we consider a transfer scenario where target data should be classified given 1-shot training samples from the same dataset and N additional samples from a heterogeneous source dataset that shares the target attribute (*e.g.*, "annual income" in Adult, Credit Risk and Electricity datasets) but contains disparate column sets (*e.g.*, "work class", "education", etc. for Adult and "loan amount", "credit history length", etc. for Credit Risk datasets).

As shown in Table 4, SPROUT consistently benefits from heterogeneous data sources (*e.g.*, 74.67%→80.00% on the Adult dataset, when $N = 10$ additional samples from the Credit Risk dataset is provided). More importantly, SPROUT is the only method that shows steady performance improvements as the number of heterogeneous training sample N increases, while the baselines are not able to properly learn from the additional samples, and their performance could even deteriorate compared to their 1-shot ($N = 0$) performances. Interestingly, cramming the extra columns from the heterogeneous dataset in LIFT-ICL only incurred noise to the accuracy. We attribute this to that our LLM-friendly descriptions enable the LLMs to exploit the deeper relationship between heterogeneous data, while the naïve concatenation without proper descriptions only perplexes the LLMs.

F. Baseline details

In this section, we provide brief explanations of the chosen baselines. For CatBoost (Prokhorenkova et al., 2018) and logistic regression, we employ the default hyperparameters as provided by the CatBoost library and the Scikit-learn library, respectively. For VIME (Yoon et al., 2020) pre-training, we adopt the optimal hyperparameters recommended in the original paper, utilizing the Adam optimizer with a learning rate of $1e-3$ and weight decay of $1e-4$. When implementing STUNT (Nam et al., 2023), we follow the unsupervised validation scheme proposed in the original paper for hyperparameter search and early stopping. For LIFT-ICL (Dinh et al., 2022), we use the generic serialization, used in the original paper, for all the datasets under consideration. Prompt example used for LIFT on the Breast dataset is provided in Table 5.

Table 5. Prompt example used for LIFT-ICL (Dinh et al., 2022) on the Breast dataset.

```
Question:When x1 is 50-59, x2 is premeno, x3 is 50-54, x4 is 0-2, x5 is yes, x6 is
2, x7 is right, x8 is left_up, x9 is yes, then what is y value?  You must choose in
[recurrence-events, no-recurrence-events].  Answer:no-recurrence-events
Question:When x1 is 40-49, x2 is premeno, x3 is 15-19, x4 is 0-2, x5 is yes, x6 is
3, x7 is right, x8 is left_up, x9 is no, then what is y value?  You must choose in
[recurrence-events, no-recurrence-events].  Answer:recurrence-events
Question:When x1 is 40-49, x2 is premeno, x3 is 15-19, x4 is 12-14, x5 is no, x6 is
3, x7 is right, x8 is right_low, x9 is yes, then what is y value?  You must choose in
[recurrence-events, no-recurrence-events].  Answer:
```

G. Experimental details

Table 6. Number of unlabeled dataset of SPROUT.

# shot	Breast	TAE	Vehicle	Hamster	Customers	LED	Pollution	Diabetes	Car
1	30	30	20	30	30	30	20	30	40
5	20	30	10	20	30	30	10	30	40

In this section, we describe the experimental details. To begin with, we have set $|\mathcal{S}| = 1$ and $|\mathcal{Q}| = 10$ across all datasets during discovering the generic indicators, as discussed in Appendix C. Further, the number of unlabeled data for each dataset used in SPROUT (*i.e.*, $|\mathcal{I}_u|$), is provided in Table 6.

Additionally, we have conducted experiments on some datasets (*i.e.*, Vehicle, Customers, LED) utilizing GPT-4 (OpenAI, 2023). This choice was made due to the limited prompt size of ChatGPT, as denoted in Table 1. However, considering the high cost of the GPT-4 API, we have considered to evaluate multiple queries within a single prompt (*i.e.*, batch-wise queries). This is accomplished by initiating each test query prompt with the prefix "`Question N`" and requesting the model to respond in the format "`Answer N`." We adopt a batch size of 30, if all samples fit within a single prompt. If not, we utilize the largest possible batch size. We also note that batching test samples in this manner represents an effective and cost-efficient method of employing SPROUT. For further clarification, a simplified prompt example, applied to the Customer dataset (*i.e.*, two unlabeled samples, one labeled sample per class, two batch queries), is illustrated in Table 7.

Table 7. Prompt example used for batch-wise test using GPT-4 on the Customers dataset.

Read a given information and questions. Think step by step, and then predict whether its value is class1 or class2. You must choose in [class1, class2]. Class1 indicates Horeca (Hotel, Restaurant, Cafe) channel, and class2 indicates Retail channel.

The dataset consists of 7 input variables: annual spending on fresh product, annual spending on milk products, annual spending on grocery products, annual spending on frozen products, annual spending on detergents and paper products, annual spending on delicatessen products, and customer's region. The output variable is the customer's channel.

Question: If the annual spending on fresh product is 1479.0, annual spending on milk products is 14982.0, annual spending on frozen products is 662.0, annual spending on detergents and paper products is 3891.0, annual spending on delicatessen products is 3508.0, customer's region (1 indicates Lispon, 2 indicates Porto, and 3 indicates Other) is 2, then what is the annual spending on grocery products? Answer: 11924.0

Question: If the annual spending on fresh product is 243.0, annual spending on milk products is 12939.0, annual spending on frozen products is 799.0, annual spending on detergents and paper products is 3909.0, annual spending on delicatessen products is 211.0, customer's region (1 indicates Lispon, 2 indicates Porto, and 3 indicates Other) is 2, then what is the annual spending on grocery products? Answer: 8852.0

Question: If the annual spending on fresh product is 918.0, annual spending on milk products is 20655.0, annual spending on grocery products is 13567.0, annual spending on frozen products is 1465.0, annual spending on detergents and paper products is 6846.0, annual spending on delicatessen products is 806.0, customer's region (1 indicates Lispon, 2 indicates Porto, and 3 indicates Other) is 2, then what is the customer's channel? Choose between [class1, class2]. Class1 indicates Horeca (Hotel, Restaurant, Cafe) channel, and class2 indicates Retail channel. Answer: class2

Question: If the annual spending on fresh product is 3097.0, annual spending on milk products is 4230.0, annual spending on grocery products is 16483.0, annual spending on frozen products is 575.0, annual spending on detergents and paper products is 241.0, annual spending on delicatessen products is 2080.0, customer's region (1 indicates Lispon, 2 indicates Porto, and 3 indicates Other) is 3, then what is the customer's channel? Choose between [class1, class2]. Class1 indicates Horeca (Hotel, Restaurant, Cafe) channel, and class2 indicates Retail channel. Answer: class1

Question 1: If the annual spending on fresh product is 11686.0, annual spending on milk products is 2154.0, annual spending on grocery products is 6824.0, annual spending on frozen products is 3527.0, annual spending on detergents and paper products is 592.0, annual spending on delicatessen products is 697.0, customer's region (1 indicates Lispon, 2 indicates Porto, and 3 indicates Other) is 1, then what is the customer's channel? Choose between [class1, class2]. Class1 indicates Horeca (Hotel, Restaurant, Cafe) channel, and class2 indicates Retail channel.

Question 2: If the annual spending on fresh product is 2083.0, annual spending on milk products is 5007.0, annual spending on grocery products is 1563.0, annual spending on frozen products is 1120.0, annual spending on detergents and paper products is 147.0, annual spending on delicatessen products is 1550.0, customer's region (1 indicates Lispon, 2 indicates Porto, and 3 indicates Other) is 1, then what is the customer's channel? Choose between [class1, class2]. Class1 indicates Horeca (Hotel, Restaurant, Cafe) channel, and class2 indicates Retail channel.

(Answer the questions in the format "Answer i: (answer)", starting from Answer 1.)

H. Dataset details

Table 8. Dataset description. We select 9 tabular datasets from the OpenML repository (Vanschoren et al., 2014) for extensive evaluation.

Property \ Dataset	Breast	TAE	Vehicle	Hamster	Customers	LED	Pollution	Diabetes	Car
OpenML id	13	48	54	893	1511	40496	882	37	40975
# Columns	9	5	18	5	7	7	15	8	6
# Numerical	0	1	18	5	6	0	15	8	0
# Categorical	9	4	0	0	1	7	0	0	6
# Classes	2	3	4	2	2	10	2	2	4

In this section, we provide detailed descriptions of the considered datasets chosen from the OpenML repository (Vanschoren et al., 2014). We select six tabular datasets (*i.e.*, Breast, TAE, Vehicle, Hamster, Customers, LED) which have been previously used in the in-context learning experiments by Dinh et al. (2022). Additionally, we incorporate three other datasets from the OpenML repository (*i.e.*, Pollution, Diabetes, Car) to verify our method across diverse types of tabular datasets. The Pollution dataset, for instance, consists of only numerical features. Likewise, the Diabetes dataset, widely acknowledged as one of the most frequently utilized datasets in tabular learning literature (Hegselmann et al., 2023; Nam et al., 2023), also consists of only numerical features. Contrasting with the previous two, the Car dataset is composed solely of categorical features. We provide detailed dataset description in Table 8.

I. Prompt examples used in SPROUT

In this section, we provide examples of prompts used in SPROUT, specifically focusing on the Breast dataset and the Customers dataset. In particular, we illustrate the prompts employed to identify the column feature with the highest correlation (see Table 9 and Table 10), along with the prompts used during the final inference stage (see Table 11 and Table 12). For the sake of brevity and due to constraints on paper length, the prompts we provide consist of merely two unlabeled samples along with a single labeled sample per class. Furthermore, we present a prompt wherein the language descriptions have been completely omitted—that is, employing generic indicators and substituting categorical features with random alphabetical symbols (see Table 13 and Table 14).

Table 9. Prompt example when identifying the highest correlated feature on the Breast dataset.

```
Question:When age is 50-59, menopause is premeno, tumor-size is 15-19, inv-nodes is 0-2,
node-caps is no, deg-malig is 2, breast is right, breast-quad is right_low, irradiat
is no, then what is the breast-cancer class?  You must choose in [recurrence-events,
no-recurrence-events].  Answer:no-recurrence-events

Question:When age is 30-39, menopause is premeno, tumor-size is 25-29, inv-nodes is 6-8,
node-caps is yes, deg-malig is 3, breast is left, breast-quad is right_low, irradiat
is yes, then what is the breast-cancer class?  You must choose in [recurrence-events,
no-recurrence-events].  Answer:recurrence-events

Question:Sort by input variables that are more related to the breast-cancer class.
Choices:  [age, menopause, tumor-size, inv-nodes, node-caps, deg-malig, breast,
breast-quad, irradiat].  Answer:
```

Table 10. Prompt example when identifying the highest correlated feature on the Customers dataset.

```
Question: If the annual spending on fresh product is 583.0, annual spending on milk
products is 685.0, annual spending on grocery products is 2216.0, annual spending on
frozen products is 469.0, annual spending on detergents and paper products is 954.0,
annual spending on delicatessen products is 18.0, customer's region (1 indicates Lispon, 2
indicates Porto, and 3 indicates Other) is 1, then what is the customer's channel? Choose
between [class1, class2]. Class1 indicates Horeca (Hotel, Restaurant, Cafe) channel, and
class2 indicates Retail channel. Answer: class1

Question: If the annual spending on fresh product is 7823.0, annual spending on milk
products is 6245.0, annual spending on grocery products is 6544.0, annual spending on
frozen products is 4154.0, annual spending on detergents and paper products is 4074.0,
annual spending on delicatessen products is 964.0, customer's region (1 indicates Lispon,
2 indicates Porto, and 3 indicates Other) is 3, then what is the customer's channel?
Choose between [class1, class2]. Class1 indicates Horeca (Hotel, Restaurant, Cafe)
channel, and class2 indicates Retail channel. Answer: class2

Question: Sort features that are more related to the customer's channel.Choices: [annual
spending on fresh product, annual spending on milk products, annual spending on grocery
products, annual spending on frozen products, annual spending on detergents and paper
products, annual spending on delicatessen products, customer's region]. Answer:
```

Table 11. Prompt example of SPROUT on the Breast dataset.

```
Read a given information and questions. Think step by step, and then predict
whether its value is recurrence-events or no-recurrence-events. You must choose in
[recurrence-events, no-recurrence-events].

Dataset has age, menopause, tumor-size, inv-nodes, node-caps, deg-malig, breast,
breast-quad, irradiat as 9 input variables and breast-cancer class as output.

Question:When age is 30-39, menopause is premeno, tumor-size is 15-19, inv-nodes is 6-8,
node-caps is yes, breast is left, breast-quad is left_low, irradiat is yes, then what is
deg-malig? Answer:3

Question:When age is 50-59, menopause is premeno, tumor-size is 15-19, inv-nodes is 0-2,
node-caps is no, breast is right, breast-quad is left_low, irradiat is no, then what is
deg-malig? Answer:2

Question:When age is 30-39, menopause is premeno, tumor-size is 25-29, inv-nodes is 6-8,
node-caps is yes, deg-malig is 3, breast is left, breast-quad is right_low, irradiat
is yes, then what is the breast-cancer class? You must choose in [recurrence-events,
no-recurrence-events]. Answer:recurrence-events

Question:When age is 50-59, menopause is premeno, tumor-size is 15-19, inv-nodes is 0-2,
node-caps is no, deg-malig is 2, breast is right, breast-quad is right_low, irradiat
is no, then what is the breast-cancer class? You must choose in [recurrence-events,
no-recurrence-events]. Answer:no-recurrence-events

Question:When age is 40-49, menopause is premeno, tumor-size is 15-19, inv-nodes is 12-14,
node-caps is no, deg-malig is 3, breast is right, breast-quad is right_low, irradiat
is yes, then what is the breast-cancer class? You must choose in [recurrence-events,
no-recurrence-events]. Answer:
```

Table 12. Prompt example of SPROUT on the Customers dataset.

Read a given information and questions. Think step by step, and then predict whether its value is class1 or class2. You must choose in [class1, class2]. Class1 indicates Horeca (Hotel, Restaurant, Cafe) channel, and class2 indicates Retail channel.

The dataset consists of 7 input variables: annual spending on fresh product, annual spending on milk products, annual spending on grocery products, annual spending on frozen products, annual spending on detergents and paper products, annual spending on delicatessen products, and customer's region. The output variable is the customer's channel.

Question: If the annual spending on fresh product is 3191.0, annual spending on milk products is 1993.0, annual spending on grocery products is 1799.0, annual spending on frozen products is 1730.0, annual spending on delicatessen products is 710.0, customer's region (1 indicates Lispon, 2 indicates Porto, and 3 indicates Other) is 1, then what is the annual spending on detergents and paper products? Answer: 234.0

Question: If the annual spending on fresh product is 5224.0, annual spending on milk products is 7603.0, annual spending on grocery products is 8584.0, annual spending on frozen products is 2540.0, annual spending on delicatessen products is 238.0, customer's region (1 indicates Lispon, 2 indicates Porto, and 3 indicates Other) is 3, then what is the annual spending on detergents and paper products? Answer: 3674.0

Question: If the annual spending on fresh product is 583.0, annual spending on milk products is 685.0, annual spending on grocery products is 2216.0, annual spending on frozen products is 469.0, annual spending on detergents and paper products is 954.0, annual spending on delicatessen products is 18.0, customer's region (1 indicates Lispon, 2 indicates Porto, and 3 indicates Other) is 1, then what is the customer's channel? Choose between [class1, class2]. Class1 indicates Horeca (Hotel, Restaurant, Cafe) channel, and class2 indicates Retail channel. Answer: class1

Question: If the annual spending on fresh product is 7823.0, annual spending on milk products is 6245.0, annual spending on grocery products is 6544.0, annual spending on frozen products is 4154.0, annual spending on detergents and paper products is 4074.0, annual spending on delicatessen products is 964.0, customer's region (1 indicates Lispon, 2 indicates Porto, and 3 indicates Other) is 3, then what is the customer's channel? Choose between [class1, class2]. Class1 indicates Horeca (Hotel, Restaurant, Cafe) channel, and class2 indicates Retail channel. Answer: class2

Question: If the annual spending on fresh product is 11686.0, annual spending on milk products is 2154.0, annual spending on grocery products is 6824.0, annual spending on frozen products is 3527.0, annual spending on detergents and paper products is 592.0, annual spending on delicatessen products is 697.0, customer's region (1 indicates Lispon, 2 indicates Porto, and 3 indicates Other) is 1, then what is the customer's channel? Choose between [class1, class2]. Class1 indicates Horeca (Hotel, Restaurant, Cafe) channel, and class2 indicates Retail channel. Answer:

Table 13. Generic prompt of SPROUT on the Breast dataset.

```
Read a given information and questions. Think step by step, and then predict its value.
You must choose in [classA, classB].

Dataset has 9 input variables and output.

Question: If input variable1 is A, input variable2 is C, input variable4 is G, input
variable5 is A, input variable6 is 2, input variable7 is A, input variable8 is A, input
variable9 is A, then what is input variable3? Answer: I

Question: If input variable1 is D, input variable2 is C, input variable4 is A, input
variable5 is B, input variable6 is 3, input variable7 is B, input variable8 is A, input
variable9 is B, then what is input variable3? Answer: I

Question: If input variable1 is A, input variable2 is C, input variable3 is I, input
variable4 is G, input variable5 is A, input variable6 is 2, input variable7 is A, input
variable8 is C, input variable9 is A, then what is output? You must choose in [classA,
classB]. Answer: classB

Question: If input variable1 is D, input variable2 is C, input variable3 is E, input
variable4 is A, input variable5 is B, input variable6 is 3, input variable7 is B, input
variable8 is C, input variable9 is B, then what is output? You must choose in [classA,
classB]. Answer: classA

Question: If input variable1 is B, input variable2 is C, input variable3 is I, input
variable4 is E, input variable5 is A, input variable6 is 3, input variable7 is A, input
variable8 is C, input variable9 is B, then what is output? You must choose in [classA,
classB]. Answer:
```

Table 14. Generic prompt example of SPROUT on the Customers dataset.

```
Read a given information and questions. Think step by step, and then predict its value.
You must choose in [classA, classB].

Dataset has 7 features and output y.

Question: If feature1 is 3191.0, feature2 is 1993.0, feature3 is 1799.0, feature4 is
1730.0, feature5 is 234.0, feature6 is 710.0, then what is feature7? Answer: 1

Question: If feature1 is 5224.0, feature2 is 7603.0, feature3 is 8584.0, feature4 is
2540.0, feature5 is 3674.0, feature6 is 238.0, then what is feature7? Answer: 3

Question: If feature1 is 583.0, feature2 is 685.0, feature3 is 2216.0, feature4 is 469.0,
feature5 is 954.0, feature6 is 18.0, feature7 is 1, then what is output y? You must
choose in [classA, classB]. Answer: classA

Question: If feature1 is 7823.0, feature2 is 6245.0, feature3 is 6544.0, feature4 is
4154.0, feature5 is 4074.0, feature6 is 964.0, feature7 is 3, then what is output y? You
must choose in [classA, classB]. Answer: classB

Question: If feature1 is 11686.0, feature2 is 2154.0, feature3 is 6824.0, feature4 is
3527.0, feature5 is 592.0, feature6 is 697.0, feature7 is 1, then what is output y? You
must choose in [classA, classB]. Answer:
```

J. Details on generic indicator discovery via SPROUT

Table 15. Candidates of generic indicators and selected result through SPROUT.

x	y	Breast	TAE	Vehicle	Hamster	Customers	LED	Pollution	Diabetes	Car
x	y value		✓							
feature	output y					✓	✓	✓		
input variable	output	✓			✓					
independent variable	dependent variable								✓	✓
predictor variable	response variable									
attribute	target variable			✓						

In this section, we provide further details on generic indicator discovery via SPROUT , as discussed in Appendix C. We detail the six candidates used for this purpose in Table 15. In addition, we present the generic indicators selected through our proposed methodology in Table 15.

K. Broader impacts

Tabular data often include privacy-sensitive or confidential features, such as social security numbers. As such, it is crucial to handle this data with care. However, SPROUT is also effective for managing anonymized features. For instance, our experiments indicate that even when categorical features are replaced with random alphabetical symbols, and generic indicators are used instead of actual column names, SPROUT still shows competitive performance. Therefore, despite potential privacy concerns related to tabular classification, SPROUT shows promise for widespread use alongside privacy-preserving techniques.

L. Experimental details on the knowledge transfer in heterogeneous data

Table 16. Dataset descriptions for the knowledge transfer experiments.

Property \ Dataset	Adult	Credit-g	Credit Risk	Electricity	Credit Approval
OpenML id	1590	31	43454	43588	29
# Columns	14	20	11	8	15
# Numerical	6	8	8	8	6
# Categorical	8	12	3	0	9
# Classes	2	2	2	2	2

In this section, we provide additional experimental details for the knowledge transfer scenario between heterogeneous data sources in Appendix E. As presented in Table 4, we consider two target datasets (Adult and Credit-g) and three source datasets (Credit Risk, Electricity, and Credit Approval) from the OpenML repository (Vanschoren et al., 2014), and we provide their detailed properties and OpenML ids in Table 16. Finally, we provide an example prompt, the transfer scenario from the Credit Risk dataset to the Adult dataset in Table 17.

Table 17. Prompt example of the transfer from Credit Risk to Adult datasets.

```
Read a given information and questions.  Think step by step, and then predict the annual
income based on the given attributes.  You must choose in [class1, class2].  Class1
indicates less than 50k and class2 indicates more than 50K a year.

Question:  When person_age is 42, person_home_ownership is RENT, person_emp_length is
3.0, loan_intent is DEBTCONSOLIDATION, loan_grade is A, loan_amnt is 2575, loan_int_rate
is 6.76, loan_status is 0, loan_percent_income is 0.03, cb_person_default_on_file is N,
cb_person_cred_hist_length is 11, then what is the annual income?  (class1 indicates less
than 50k and class2 indicates more than 50K a year) Answer:  class2

Question:  When person_age is 25, person_home_ownership is OWN, person_emp_length is
1.0, loan_intent is PERSONAL, loan_grade is A, loan_amnt is 3000, loan_int_rate is
9.63, loan_status is 0, loan_percent_income is 0.1, cb_person_default_on_file is N,
cb_person_cred_hist_length is 2, then what is the annual income?  (class1 indicates less
than 50k and class2 indicates more than 50K a year) Answer:  class1

Question:  When person_age is 27, person_home_ownership is MORTGAGE, person_emp_length
is 2.0, loan_intent is VENTURE, loan_grade is B, loan_amnt is 16000, loan_int_rate is
12.21, loan_status is 0, loan_percent_income is 0.13, cb_person_default_on_file is N,
cb_person_cred_hist_length is 5, then what is the annual income?  (class1 indicates less
than 50k and class2 indicates more than 50K a year) Answer:  class2

Question:  When person_age is 25, person_home_ownership is RENT, person_emp_length
is 6.0, loan_intent is VENTURE, loan_grade is A, loan_amnt is 4000, loan_int_rate is
6.39, loan_status is 0, loan_percent_income is 0.14, cb_person_default_on_file is N,
cb_person_cred_hist_length is 3, then what is the annual income?  (class1 indicates less
than 50k and class2 indicates more than 50K a year) Answer:  class1

Question:  When person_age is 32, person_home_ownership is RENT, person_emp_length
is 16.0, loan_intent is VENTURE, loan_grade is B, loan_amnt is 6300, loan_int_rate is
9.91, loan_status is 0, loan_percent_income is 0.13, cb_person_default_on_file is N,
cb_person_cred_hist_length is 6, then what is the annual income?  (class1 indicates less
than 50k and class2 indicates more than 50K a year) Answer:  class1

Question:  When person_age is 24, person_home_ownership is RENT, person_emp_length is
8.0, loan_intent is VENTURE, loan_grade is A, loan_amnt is 10000, loan_int_rate is
9.32, loan_status is 1, loan_percent_income is 0.42, cb_person_default_on_file is N,
cb_person_cred_hist_length is 4, then what is the annual income?  (class1 indicates less
than 50k and class2 indicates more than 50K a year) Answer:  class1

Question:  When age is 35, workclass is Private, fnlwgt is 29874.0, education is
Some-college, education-num is 10, marital-status is Married-civ-spouse, occupation is
Handlers-cleaners, relationship is Husband, race is White, sex is Male, capital-gain is
0.0, capital-loss is 0.0, hours-per-week is 40, native-country is United-States, then what
is the annual income?  (class1 indicates less than 50k and class2 indicates more than 50K
a year) Answer:  class1

Question:  When age is 50, workclass is Self-emp-not-inc, fnlwgt is 29231.0, education
is HS-grad, education-num is 9, marital-status is Married-civ-spouse, occupation is
Exec-managerial, relationship is Husband, race is White, sex is Male, capital-gain is 0.0,
capital-loss is 0.0, hours-per-week is 45, native-country is United-States, then what is
the annual income?  (class1 indicates less than 50k and class2 indicates more than 50K a
year) Answer:  class2

Question:  When age is 53, workclass is Self-emp-not-inc, fnlwgt is 169112.0, education
is Bachelors, education-num is 13, marital-status is Married-civ-spouse, occupation is
Exec-managerial, relationship is Husband, race is White, sex is Male, capital-gain is 0.0,
capital-loss is 0.0, hours-per-week is 40, native-country is Hungary, then what is the
annual income?  (class1 indicates less than 50k and class2 indicates more than 50K a year)
Answer:
```