Supplementary Material: STiL: Semi-supervised Tabular-Image Learning for Comprehensive Task-Relevant Information Exploration in Multimodal Classification

Siyi Du[1⋆] Xinzhe Luo[1] Declan P. O'Regan[2] Chen Qin[1⋆]
[1]Department of Electrical and Electronic Engineering & I-X, [2]MRC Laboratory of Medical Science
Imperial College London, London, UK
{s.du23, x.luo, declan.oregan, c.qin15}@imperial.ac.uk

A. Detailed Formulations

Disentanglement Loss: Our proposed disentanglement loss (Sec 3.2 of the manuscript) aims to minimize mutual information (MI) between modality-shared and modality-specific representations. As MI is intractable, we leverage an upper bound called the contrastive log-ratio upper bound (CLUB) [4, 19] as an MI estimator. Given sample pairs $\{(\boldsymbol{a}_j, \boldsymbol{b}_j)\}_j^N$, CLUB is defined as:

$$\begin{aligned} I_{CLUB}(\boldsymbol{a}, \boldsymbol{b}) &= \mathbb{E}_{p(\boldsymbol{a},\boldsymbol{b})}[\log p(\boldsymbol{b}|\boldsymbol{a})] - \mathbb{E}_{p(\boldsymbol{a})}\mathbb{E}_{p(\boldsymbol{b})}[\log p(\boldsymbol{b}|\boldsymbol{a})] \\ &= \frac{1}{N}\sum_{j}^{N} \log p(\boldsymbol{b}_j|\boldsymbol{a}_j) - \frac{1}{N^2}\sum_{j=1}^{N}\sum_{k=1}^{N} \log p(\boldsymbol{b}_k|\boldsymbol{a}_j) \\ &= \frac{1}{N^2}\sum_{j=1}^{N}\sum_{k=1}^{N} [\log p(\boldsymbol{b}_j|\boldsymbol{a}_j) - \log p(\boldsymbol{b}_k|\boldsymbol{a}_j)], \end{aligned} \tag{S1}$$

where $\log p(\boldsymbol{b}_j|\boldsymbol{a}_j)$ denotes the conditional log-likelihood of a positive sample pair $(\boldsymbol{a}_j, \boldsymbol{b}_j)$, and $\{\log p(\boldsymbol{b}_k|\boldsymbol{a}_j)\}_{j\neq k}$ is the conditional log-likelihood of a negative sample pair $(\boldsymbol{a}_j, \boldsymbol{b}_k)$. However, as STiL obtains modality-shared and modality-specific representations simultaneously during training, the exact conditional distribution between these two representations is unavailable. To address this limitation, similar to [4, 19], we leverage a variational distribution $q_\theta(\boldsymbol{b}|\boldsymbol{a})$ (an MLP layer with parameter θ) to approximate $p(\boldsymbol{b}|\boldsymbol{a})$. This leads to a variational CLUB (vCLUB), formulated as:

$$\begin{aligned} I_{vCLUB}(\boldsymbol{a}, \boldsymbol{b}) &= \mathbb{E}_{p(\boldsymbol{a},\boldsymbol{b})}[\log q_\theta(\boldsymbol{b}|\boldsymbol{a})] - \mathbb{E}_{p(\boldsymbol{a})}\mathbb{E}_{p(\boldsymbol{b})}[\log q_\theta(\boldsymbol{b}|\boldsymbol{a})] \\ &= \frac{1}{N}\sum_{j}^{N} \log q_\theta(\boldsymbol{b}_j|\boldsymbol{a}_j) - \frac{1}{N^2}\sum_{j=1}^{N}\sum_{k=1}^{N} \log q_\theta(\boldsymbol{b}_k|\boldsymbol{a}_j) \\ &= \frac{1}{N^2}\sum_{j=1}^{N}\sum_{k=1}^{N} [\log q_\theta(\boldsymbol{b}_j|\boldsymbol{a}_j) - \log q_\theta(\boldsymbol{b}_k|\boldsymbol{a}_j)]. \end{aligned} \tag{S2}$$

To enforce $q_\theta(\boldsymbol{b}|\boldsymbol{a})$ align closely with $p(\boldsymbol{b}|\boldsymbol{a})$, we maximize the following log-likelihood:

$$\mathcal{L}_{q_\theta}(\boldsymbol{a}, \boldsymbol{b}) = \frac{1}{N}\sum_{j}^{N} \log q_\theta(\boldsymbol{b}_j|\boldsymbol{a}_j). \tag{S3}$$

Finally, our disentanglement losses $\mathcal{L}_{ds}^i$ and $\mathcal{L}_{ds}^t$ can be formulated as:

$$\mathcal{L}_{ds}^i = I_{vCLUB}(\boldsymbol{z}_c^i, \boldsymbol{z}_s^i) - \mathcal{L}_{q_\theta}(\boldsymbol{z}_c^i, \boldsymbol{z}_s^i) \tag{S4}$$

$$\mathcal{L}_{ds}^t = I_{vCLUB}(\boldsymbol{z}_c^t, \boldsymbol{z}_s^t) - \mathcal{L}_{q_\theta}(\boldsymbol{z}_c^t, \boldsymbol{z}_s^t), \tag{S5}$$

where $\boldsymbol{z}_c^i$ and $\boldsymbol{z}_c^t$ are modality-specific representations and $\boldsymbol{z}_s^i$ and $\boldsymbol{z}_s^t$ are modality-shared representations. These two losses are used in Eq. (3) of the manuscript.

B. Implementation Details

Datasets: The UK Biobank (UKBB) dataset [14] consists of magnetic resonance images (MRIs) and tabular data related to cardiac diseases. Following prior work [5, 8], we used mid-ventricle slices from cardiac MRIs in three time phases, *i.e.*, end-systolic (ES) frame, end-diastolic (ED) frame, and an intermediate time frame between ED and ES. In addition, we employed 75 disease-related tabular features, including 26 categorical features (*e.g.*, alcohol drinker status) and 49 continuous features (*e.g.*, average heart rate). The DVM dataset [10] includes 2D RGB car images along with tabular data describing the characteristics of the car. As done in [5, 8], we employed 17 tabular features, including 4 categorical features (*e.g.*, color), and 13 continuous features (*e.g.*, width). Detailed benchmark information can be found in the supplementary material of [5].

To construct a training dataset with 10% labeled samples, we randomly sampled 10% of the labeled instances from each class, ensuring that the class distribution remains consistent with the original training dataset. A similar procedure is followed when creating the 1% labeled dataset.

⋆Corresponding authors.

Table S1. Definitions of symbols used for STiL's hyper-parameters.

	Description
B	Batch size of labeled data
μ	Relative size ratio between labeled and unlabeled batches
α	Weighting coefficient controlling the labeled cross-entropy loss $\mathcal{L}_{ce}$
β	Weighting coefficient controlling the contrastive consistency loss $\mathcal{L}_{cc}$
γ	Weighting coefficient controlling the disentanglement losses $\mathcal{L}_{ds}^{i}$ and $\mathcal{L}_{ds}^{t}$
λ_p	Weighting coefficient controlling the prototypical contrastive loss $\mathcal{L}_{pt}$
λ_u	Weighting coefficient controlling the unlabeled cross-entropy loss $\mathcal{L}_{uce}$
τ	Threshold for defining confident pseudo-labels
r	Smoothness Weighting coefficient in PGLS
m	Momentum coefficient for EMA
κ	Temperature parameter

Table S2. Hyper-parameter settings for STiL.

Task	B	μ	α	β	γ	λ_p	λ_u	τ	r	m	κ
DVM	64	7	0.2	3	0.5	1	0.2	0.9	0.9	0.996	0.1
1% CAD 10% CAD	32	7	0.2	0.5	5	0.5	5	0.85 0.8	0.95	0.4	0.1
1% Infa. 10% Infa.	32	7	0.2	1	1	0.5	2	0.85 0.8	0.95	0.4	0.1

We adopted the same data augmentation technique as described in [5, 8]. For image data, we employed random scaling, rotation, shifting, flipping, Gaussian noise, as well as brightness, saturation, and contrastive changes, followed by resizing the images to 128×128. For tabular data, categorical values (*e.g.*, 'yes', 'no', and 'blue') were converted into ordinal numbers, while continuous (numerous) values were standardized using z-score normalization. To enhance data diversity, we randomly replaced 30% of the tabular values for each subject with random values from the respective columns. Note that tabular SSL models (SCARF [1] and SAINT [13]) implement their own tabular augmentation strategies. The hyper-parameters and training configurations for the supervised and SSL models were consistent with those used in [5, 8]. The batch size was set to 512 for DVM and 256 for both CAD and Infarction. The hyper-parameter settings for the proposed STiL and SemiSL algorithms are detailed below.

The Proposed STiL: We used ResNet-50 as the image encoder and a transformer-based tabular encoder proposed by Du *et al*. [5], both initialized with publicly available pre-trained weights from [5]. The tabular encoder consists of 4 transformer layers, each with 8 attention heads and a hidden dimension of 512. For a fair comparison, all SemiSL methods used the same pre-trained encoders. Details of STiL's hyper-parameters and their configurations are provided in Tab. S1 and Tab. S2, respectively. Based on validation performance, we set the starting pseudo-labeling epoch to 25 for 10% labeled DVM, 35 for 1% labeled DVM, and 8 for both CAD and Infarction. The GFLOPS for STiL is 3.63.

CoMatch [12]: This framework relies on strong-to-weak consistency regularization and contrastive learning. It refines pseudo-labels by incorporating information from nearby samples in the embedding space, and then uses these pseudo-labels to regulate the structure of embeddings via graph-based contrastive learning. Following the original paper, we set the weight factors for unlabeled classification loss and contrastive loss, λ_{cls} and λ_{ctr}, to 10. The smoothness parameter α was set to 0.9, the embedding memory bank size K to 2,560, the temperature parameter to 0.1, and the EMA momentum to 0.996. The batch sizes for the labeled and unlabeled data were the same as those used in STiL. Additionally, based on validation performance, we set the thresholds for strong-to-weak consistency and graph-based contrastive learning as follows: $\tau = 0.8$ and $T = 0.6$ for DVM, and $\tau = 0.6$ and $T = 0.3$ for both CAD and Infarction. The starting pseudo-labeling epoch was 10 for DVM and 8 for CAD and Infarction.

CoMatch^M: This model is an extension of CoMatch to the multimodal image-tabular setting. Its hyper-parameters were the same as those in CoMatch. Based on validation performance, we set the thresholds for strong-to-weak consistency and graph-based contrastive learning as follows: $\tau = 0.9$ and $T = 0.8$ for DVM, and $\tau = 0.85$ and $T = 0.7$ for CAD and Infarction.

SimMatch [20]: This algorithm applies strong-to-weak consistency regularization at both the semantic and instance levels. It encourages different augmented views of the same instance to have the same class prediction and maintain similar similarity relationships with respect to other instances. Following the original paper, we set the weight factors for the unlabeled classification loss and the instance consistency regularization loss, *i.e.*, λ_u and λ_{in}, to 10 and 5, respectively. The smoothness parameter α was set to 0.9, the temperature parameter to 0.1, and the EMA momentum to 0.996. The batch sizes for labeled and unlabeled data were the same as those used in STiL. Based on validation performance, we set the threshold in strong-to-weak consistency regularization to 0.8 for DVM and to 0.6 for CAD and Infarction. The starting pseudo-labeling epoch was 10 for DVM and 8 for CAD and Infarction.

SimMatch^M: This model is an adaptation of SimMatch to the multimodal image-tabular setting. The hyper-parameters were the same as those used in SimMatch. According to validation performance, we set the threshold for strong-to-weak consistency regularization τ to 0.9 for DVM and to 0.85 for CAD and Infarction.

FreeMatch [17]: This approach is also based on strong-to-weak consistency regularization and focuses on effectively leveraging unlabeled data. It adjusts the confident threshold in a self-adaptive manner according to the model's learning progress. Following the original paper, we set the weight factors for unlabeled classification loss and self-adaptive fairness loss, *i.e.*, w_u and w_f, to 1 and 0.001, respectively.

Table S3. Number of parameters and learning rates for DVM, CAD, and Infarction across different algorithms. We provide the number of parameters used during both training and testing. For SSL methods, the learning rates are reported for both linear-probing (L), where the feature extractors are frozen and only the linear classifiers of the pre-trained models are tuned, and full fine-tuning (F), where all parameters are trainable. Learning rates are indicated as (L / F). "M" denotes millions, and "1e-3" represents 1×10^{-3}.

Model	Modality		DVM		CAD & Infarction	
	I	T	#Params (train/test)	learning rate	#Params (train/test)	learning rate
(a) Supervised Methods						
ResNet-50 [9]	√		24.1M / 24.1M	3e-4	23.5M / 23.5M	1e-3
DAFT [18]	√	√	26.0M / 26.0M	3e-4	25.4M / 25.4M	3e-3
IF [6]	√	√	26.9M / 26.9M	3e-4	26.3M / 26.3M	3e-3
TIP [5] w/o SSL	√	√	54.2M / 54.2M	3e-4	54.1M / 54.1M	3e-3
(b) SSL Pre-training Methods (L / F)						
SimCLR [3]	√		28.0M / 24.1M	1e-3 / 1e-4	28.0M / 23.5M	1e-3 / 1e-3
BYOL [7]	√		70.1M / 24.1M	1e-3 / 1e-4	70.1M / 23.5M	1e-3 / 1e-4
SCARF [1]		√	0.6M / 0.4M	1e-4 / 1e-4	0.7M / 0.3M	1e-3 / 1e-3
SAINT [13]		√	6.5M / 6.5M	1e-4 / 1e-5	99.1M / 99.1M	1e-3 / 1e-5
MMCL [8]	√	√	36.8M / 24.1M	1e-3 / 1e-3	36.9M / 23.5M	1e-3 / 1e-3
TIP [5]	√	√	58.8M / 54.2M	1e-4 / 1e-4	58.9M / 54.1M	1e-3 / 1e-4
(c) SemiSL Methods						
CoMatch [12]	√		28.6M / 24.1M	1e-4	28.0M / 23.5M	1e-3
SimMatch [20]	√		28.6M / 24.1M	1e-4	28.0M / 23.5M	1e-3
FreeMatch [17]	√		28.6M / 24.1M	1e-4	28.0M / 23.5M	1e-3
CoMatch^M	√	√	38.1M / 37.5M	1e-4	37.8M / 37.3M	1e-3
SimMatch^M	√	√	38.1M / 37.5M	1e-4	37.8M / 37.3M	1e-3
FreeMatch^M	√	√	38.1M / 37.5M	1e-4	37.8M / 37.3M	1e-3
Co-training [2]	√	√	38.1M /38.1M	1e-4	37.4M / 37.4M	1e-3
MMatch [15]	√	√	38.1M / 38.1M	1e-4	37.4M / 37.4M	1e-3
Self-KD [16]	√	√	44.0M / 44.0M	1e-4	43.6M / 43.6M	1e-3
STiL	√	√	46.7M / 43.0M	1e-4	46.2M / 42.0M	1e-3

The temperature parameter was set to 0.1, and the EMA momentum to 0.996. The batch sizes for the labeled and unlabeled data were the same as those used in STiL.

FreeMatch^M: This model is an extension of FreeMatch to the multimodal image-tabular setting. Its hyper-parameters were the same as those used in FreeMatch.

Co-training [2]: We adapt this co-pseudo-labeling based method to the multimodal image-tabular domain. The predictions from the image classifier serve as the pseudo-labels for the tabular classifier, and vice versa. Then a multimodal classifier trained on labeled data is used for classification. The weight factors for the labeled and unlabeled classification losses, *i.e.*, α and λ_u, as well as the EMA momentum, were the same as those used in STiL.

MMatch [15]: In MMatch, predictions from a multimodal classifier are used as pseudo-labels for training unimodal classifiers. In addition, similar to CoMatch, MMatch refines the pseudo-labels by aggregating label information from nearby samples in the embedding space. Following the original paper, we set the smoothness parameter to 0.9, and the embedding memory bank size to 640. Based on validation performance, the weight factor for the unlabeled classification loss was set to 0.2.

Self-KD [16]: This method is based on co-pseudo-labeling and cross-modal consistency regularization. In Self-KD, a multimodal classifier serves as the teacher for unimodal classifiers, transferring knowledge to them through pseudo-labeling. Meanwhile, the average ensemble of unimodal classifiers is used as the pseudo-label for training the multimodal classifier. Following the original paper, we set the weight factors for the knowledge distillation loss, the contrastive loss, and the L1-norm regularization term, *i.e.*, γ, δ, and η, to 0.6, 1, and 0.1, respectively.

The learning rate and the number of parameters for each algorithm are summarized in Tab. S3. We used the Adam optimizer [11] without weight decay and deployed all models on 2 A5000 GPUs. To mitigate overfitting, similar to [5, 8], we employed an early stopping strategy in Pytorch Lightning, with a minimal delta (divergence threshold) of 0.0001, a maximal number of epochs of 500, and a patience (stopping threshold) of 100 epochs. We ensured that all methods had converged under this training configuration.

C. Additional Experiment

Experiments with a Finer Grid of Label Percentage: In Tab. 2 and Tab. 3 of the manuscript, we compared STiL with SOTA SemiSL methods using experiments with 1%

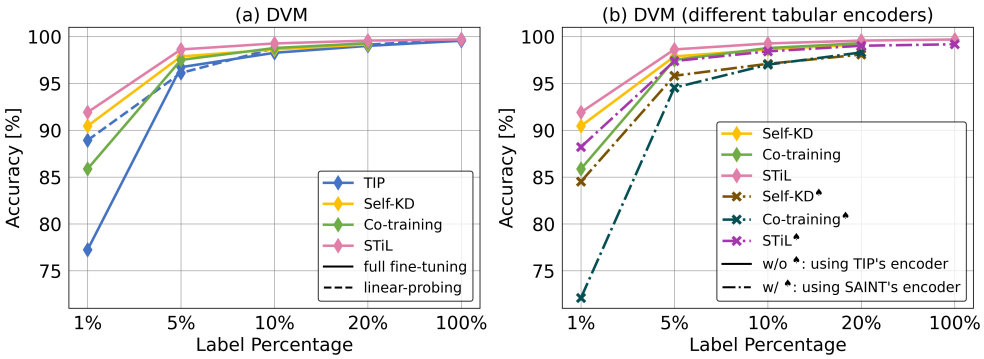

Figure S1. (a) Results comparing SSL and SemiSL multimodal SOTAs with STiL using a finer grid of label percentage. (b) Results of SemiSL multimodal SOTAs and STiL using different tabular encoders.

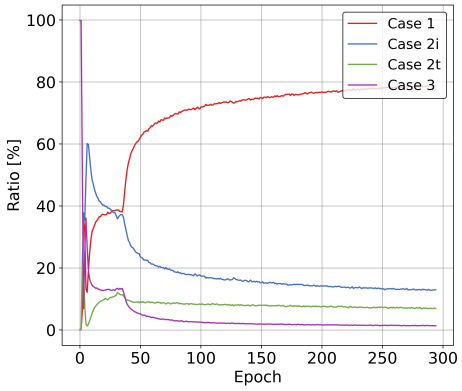

Figure S2. Sample ratios for each case in CGPL during training. The model is trained on 1% labeled DVM.

and 10% labeled samples. To provide a more detailed analysis, we further conducted experiments on DVM with additional label percentages of 5%, 20%, and 100%, as shown in Fig. S1(a). The results demonstrate that STiL consistently outperforms SOTA SSL/SemiSL methods across different label percentages.

Applicability to Different Tabular Encoders: To demonstrate the general applicability of STiL, we evaluated its performance with different tabular encoders. Specifically, we replaced TIP's pre-trained tabular encoder with SAINT [13]'s pre-trained tabular encoder. As shown in Fig. S1(b), all SemiSL approaches exhibit performance drops when using SAINT's encoder, indicating that TIP is a more powerful tabular encoder than SAINT, as also noted in TIP's paper [5]. However, while Self-KD and Co-training experience a significant performance decrease, STiL remains more stable and continues to achieve the best performance, demonstrating its robustness across different tabular encoders.

Sample Ratios for Different Cases in CGPL: As mentioned in Sec 3.3 of the manuscript, CGPL categorizes samples into 4 cases based on classifier consensus: (1) Case 1: all classifiers agree; (2) Case 2i: f^m and f^i agree; (3) Case 2t: f^m and f^t agree; and (4) Case 3: none of the above. To assess the efficacy of CGPL, we visualize the changes in the ratios of the samples belonging to each case during training. As shown in Fig. S2, the sample ratios for both case 2i and case 2t initially increase during the initial training stage but later decrease and stabilize at a lower bound. On the other hand, the sample ratio of Case 1 gradually increases and approaches an upper bound. These observations demonstrate that: (1) CGPL facilitates collaboration among classifiers, enabling them to learn from each other and improving classifier agreement; (2) due to the *Information Modality Gap*, unimodal classifiers, which rely solely on a single modality, lack comprehensive task knowledge and fail to align with the multimodal classifier on certain challenging multimodal cases; and (3) CGPL effectively generates pseudo-labels through classifiers' consensus collaboration while allowing classifier diversity, which helps reduce the risk of classifier collusion.

Class-wise Accuracy in DVM: DVM has 283 classes, each with a varying number of labeled training samples. To investigate the impact of imbalanced data on STiL and other comparing algorithms, we visualize their class-wise accuracy for both majority classes (those with more training samples) and minority classes (those with fewer training samples). Specifically, we ranked the classes based on their number of labeled training samples and displayed the class-wise accuracy for the top 16 majority classes and the bottom 16 minority classes. As shown in Fig. S3, supervised methods exhibit low accuracy across different classes, indicating their limited capacity when trained with a few labeled data. Though TIP, the SSL pre-training framework, performs well on majority classes, its accuracy significantly decreases on some minority classes. This suggests that relying solely on

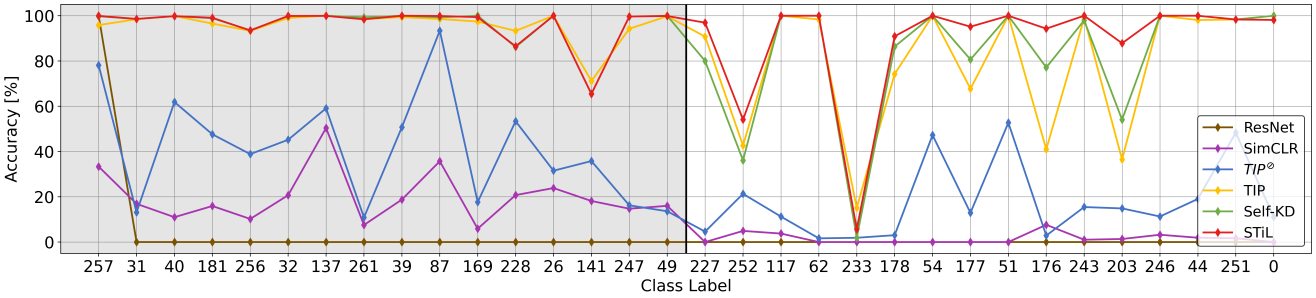

Figure S3. Class-wise accuracy comparing STiL and other methods trained on 1% labeled DVM. The top 16 majority classes are shown in the grey region, while the bottom 16 minority classes are shown in the white region. TIP$^{\oslash}$ represents TIP w/o SSL pre-training.

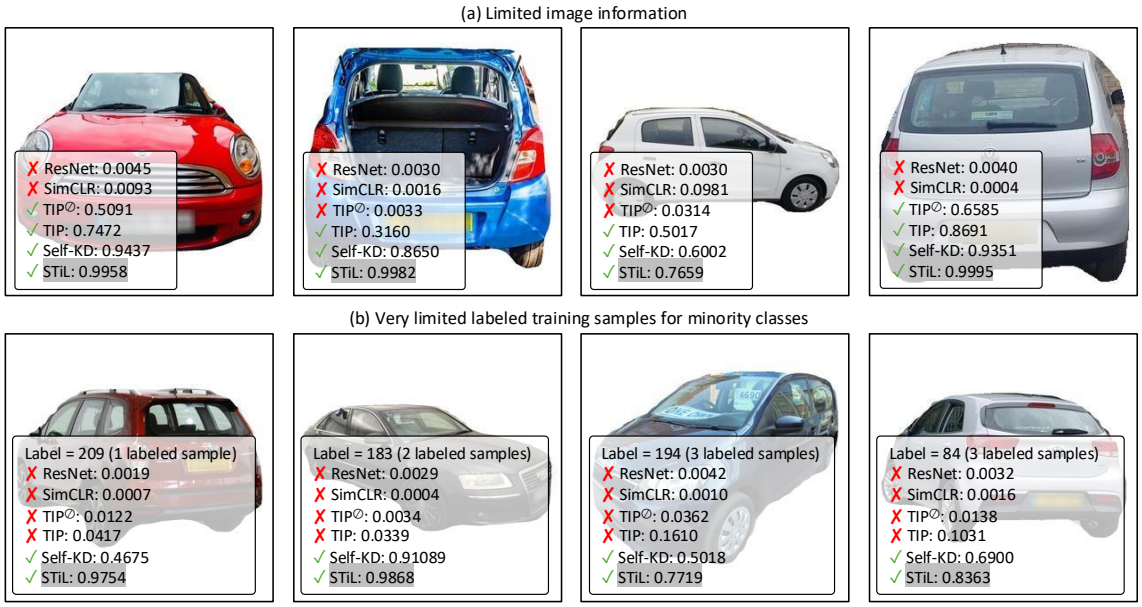

Figure S4. DVM car visualization of challenging samples and the ground-truth class predictions for STiL and other models trained on 1% labeled DVM. (a) Samples with limited image information, where the views of cars are restricted due to shooting angles (compared to the samples shown in (b)); (b) Samples from minority classes. $\times$ indicates the model predicts a wrong class, while $\surd$ indicates the model predicts the correct class.

a small amount of labeled data during fine-tuning is ineffective, especially for minority classes. In contrast, STiL mitigates these issues by leveraging labeled and unlabeled data jointly, achieving overall better results. In addition, we observe that all models perform poorly on class 233, which can be attributed to the very limited labeled data (only 1 training sample) and the inherent difficulty in classifying this class.

Case Study: We visualize several challenging examples where STiL outperforms previous SOTAs. The results show that (1) a single image modality is insufficient to solve the classification task (the failure of ResNet and SimCLR in Fig. S4(a)) and (2) minority classes with very limited labeled training samples pose challenges for SSL algorithms (the failure of TIP in Fig. S4(b)). However, STiL enables the model to comprehensively explore task-relevant information from both labeled and unlabeled data, leading to improved performance on these challenging samples.

References

[1] Dara Bahri, Heinrich Jiang, Yi Tay, and Donald Metzler. SCARF: Self-supervised contrastive learning using random feature corruption. In *ICLR*, 2022. 2, 3

[2] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *COLT*, 1998. 3

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*. PMLR, 2020. 3

[4] Pengyu Cheng, Weituo Hao, Shuyang Dai, Jiachang Liu, Zhe Gan, and Lawrence Carin. CLUB: A contrastive log-ratio

upper bound of mutual information. In *ICML*. PMLR, 2020. 1

[5] Siyi Du, Shaoming Zheng, Yinsong Wang, Wenjia Bai, Declan P. O'Regan, and Chen Qin. TIP: Tabular-image pre-training for multimodal classification with incomplete data. In *ECCV*, 2024. 1, 2, 3, 4

[6] Hongyi Duanmu, Pauline Boning Huang, Srinidhi Brahmavar, Stephanie Lin, Thomas Ren, Jun Kong, Fusheng Wang, and Tim Q Duong. Prediction of pathological complete response to neoadjuvant chemotherapy in breast cancer using deep learning with integrative imaging, molecular and demographic data. In *MICCAI*. Springer, 2020. 3

[7] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, et al. Bootstrap your own latent-a new approach to self-supervised learning. *NIPS*, 2020. 3

[8] Paul Hager, Martin J Menten, and Daniel Rueckert. Best of both worlds: Multimodal contrastive learning with tabular and imaging data. In *CVPR*, 2023. 1, 2, 3

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[10] Jingmin Huang, Bowei Chen, Lan Luo, et al. DVM-CAR: A large-scale automotive dataset for visual marketing research and applications. In *2022 IEEE International Conference on Big Data (Big Data)*. IEEE, 2022. 1

[11] Diederik P Kingma and Jimmy Ba. ADAM: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 3

[12] Junnan Li, Caiming Xiong, and Steven CH Hoi. CoMatch: Semi-supervised learning with contrastive graph regularization. In *ICCV*, 2021. 2, 3

[13] Gowthami Somepalli, Micah Goldblum, Avi Schwarzschild, C Bayan Bruss, and Tom Goldstein. SAINT: Improved neural networks for tabular data via row attention and contrastive pre-training. *arXiv preprint arXiv:2106.01342*, 2021. 2, 3, 4

[14] Cathie Sudlow, John Gallacher, Naomi Allen, Valerie Beral, Paul Burton, et al. UK Biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS Medicine*, 2015. 1

[15] Xiaoli Wang, Liyong Fu, Yudong Zhang, Yongli Wang, and Zechao Li. MMatch: Semi-supervised discriminative representation learning for multi-view classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 2022. 3

[16] Xiaoli Wang, Yongli Wang, Guanzhou Ke, Yupeng Wang, and Xiaobin Hong. Knowledge distillation-driven semi-supervised multi-view classification. *Information Fusion*, 2024. 3

[17] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, et al. FreeMatch: Self-adaptive thresholding for semi-supervised learning. In *ICLR*, 2022. 2, 3

[18] Tom Nuno Wolf, Sebastian Pölsterl, Christian Wachinger, Alzheimer's Disease Neuroimaging Initiative, et al. DAFT: a universal module to interweave tabular data and 3d images in CNNs. *NeuroImage*, 2022. 3

[19] Yilan Zhang, Yingxue Xu, Jianqi Chen, Fengying Xie, and Hao Chen. Prototypical information bottlenecking and disentangling for multimodal cancer survival prediction. In *ICLR*, 2023. 1

[20] Mingkai Zheng, Shan You, Lang Huang, Fei Wang, Chen Qian, and Chang Xu. SimMatch: Semi-supervised learning with similarity matching. In *CVPR*, 2022. 2, 3