On Transferring Expert Knowledge from Tabular Data to Images
Supplementary Material

Anonymous Author(s)
Affiliation
Address
email

A Experiment Details

A.1 Dataset Details

The datasets used in our experiments are MFEAT [13], Data Visual Marketing (DVM) [6], SUNAttribute [11], CelebA [8], PetFinder-adoption, PetFinder-pawpularity and Avito.

MFEAT. This dataset consists of features of handwritten numerals ('0'–'9') extracted from a collection of Dutch utility maps. 200 patterns per class (for a total of $2,000$ patterns) have been digitized in binary images. These digits are represented in terms of the following six feature sets. We use only 76 fourier coefficients of the character shapes and 6 morphological features for tabular data. The image modality is reconstructed from 240 pixel averages of images from 2×3 windows.

DVM. DVM dataset aims to facilitate business related research and applications in automotive industry such as car appearance design, consumer analytics and sales modeling. The dataset contains car images, model specifications and sales information about 899 car models that have been sold in the UK market over the last 20 years. which comprises two data parts: the image data and the table data. The former contains $1,451$,784 car images that have been deliberately cleaned and organized. While the latter includes six CSV tables that cover the non-visual attributes such as brand, price, sales, etc. Different from MMCL, only the new version DVM dataset is available [3]. We pair this tabular data with a single random image from each advertisement, yielding a dataset of $70,580$ train pairs, $17,645$ validation pairs, and $88,226$ test pairs. Car models with less than 700 samples were removed, resulting in 129 target classes, classification task. There are total of 13 numerical variables and 3 categorical variables in this dataset. We expect that under the guidance of tabular data, images can learn more knowledge and make classification better.

The DVM dataset utilized in the original paper is an earlier version, and unfortunately, we don't have access to the dataset after the official update. This discrepancy in dataset versions may introduce variations in the data distribution and characteristics. Specifically, all the images are resized to 300x300 resolutions; Segment results are no longer provided directly; Image data of 2019 registered car models is added and the non-visual feature data is updated to 2020.

We follow the steps in [3] in Section 4.1 to preprocess the data. In detail, the car models with less than 700 samples were removed, resulting in 129 target classes. This process ensures that the amount of data remain largely consistent with [3].

Lastly, to maintain uniformity and facilitate fair comparisons, we employed a fixed batch size of 64 across all methods, whereas the original paper employed a larger 512. Additionally, we conducted MMCL method on our dataset with a batch size of 512. The result was 0.8869/0.9070. This is still somewhat different from the values reported in [3] and performs worse than our method 0.9207 with a batch size of 512.

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.

Furthermore, we conducted a comparison of GPU usage with batch size 64. Our method uses 8 GB of memory while theirs uses 20 GB. The results revealed that the MMCL method remains resource-intensive. Conversely, our method achieves superior performance with lower computational costs, further highlighting the efficiency of our approach.

SUNAttribute. SUNAttribute annotates 20 scenes from each of the 717 SUN categories. Each scene has 102 attributes and each attribute will have multiple annotations. For simplicity, we divide each attribute into zero and one and our goal is to predict whether a scene is an open space, which is a binary classification task. The dataset contains $14,340$ images and the corresponding table feature, each attribute of the table feature represents a scene and takes the value of 1 if the attribute is present in the image. we use $8:1:1$ to divide the training set, validation set, and testing set. There are total of 101 categorical variables in this dataset.

CelebA. is the abbreviation of CelebFaces Attribute, meaning celebrity face attribute dataset, which contains $202,599$ face images of $10,177$ celebrities, each image is well marked with features, including 40 attribute markers such as Big_Nose. We use Attractive as the label, which is a binary classification task. We use $8:1:1$ to divide the training set, validation set, and testing set. There are total of 39 categorical variables in this dataset. We expect to introduce more detailed face information in the table, allowing the image to perform better on downstream tasks.

PetFinder-adoption. Animal adoption rates are strongly correlated to the metadata associated with their online profiles, such as descriptive text and photo characteristics. This dataset comes from a kaggle competition where the task is to predict the speed at which a pet is adopted, which is a five-class classification task. There are total of 10 numerical variables and 14 categorical variables in this dataset. Tabular data contains information about the pet such as the type and vaccination status. We also use the same division for the dataset.

PetFinder-pawpularity. This dataset also comes from a kaggle competition where the task was to predict the popularity of a pet based on that pet's profile and photo, which is a regression task. Each pet photo is labeled with the value of 1 (Yes) or 0 (No) for each of features. For example, "Face" represents whether the face of the pet in the picture is frontal. There are 12 categorical variables in tabular data.

Avito. Avito, Russia's largest classified advertisements website, is deeply familiar with this problem. Sellers on their platform sometimes feel frustrated with both too little demand (indicating something is wrong with the product or the product listing) or too much demand (indicating a hot item with a good description was underpriced). This dataset is challenging you to predict demand for an online advertisement based on its full description, its context and historical demand for similar ads in similar contexts. The target deal_probability can be any float from zero to one. It's also a regression task. There are total of 2 numerical variables such as and 11 categorical variables such as in this dataset.

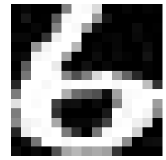
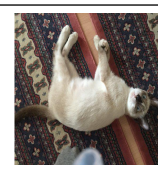
A.2 Training Details

We use ResNet50 with weight pretrained on ImageNet-1k [12] as image feature extractor for all methods mentioned in this paper. The classifier is built from an MLP with one hidden layer of size 1024.

For baseline methods, the numerical tabular data fields are standardized using z-score normalization with a mean value of 0 and standard deviation of 1. For our method CHARMS, we use FT-Transformer [2] to get the embedding of tabular data, which can process continuous and categorical variables separately.

- **KD [5]:** For KD method, we search the temperatures in $\{1.0, 2.0, 4.0, 6.0, 8.0\}$ and λ in $\{0.2, 0.4, 0.6, 0.8\}$.
- **KD-Fou:** This means that we use only 76 fourier coefficients of the character shapes features when training the teacher network.
- **KD-Mor:** This means that we use only 6 morphological features when training the teacher network, which can be revealed in images.
- **FMR [17]:** We set ten percent of the fixed features to be knockdown in each epoch in FMR method. The fixed feature classifier is a linear connection between tabular data and the corresponding image.

Table 1: Introduction to the dataset. Here we introduce image data and tabular data in each dataset, and numerical and categorical variables are introduced separately in the tabular data. An example is given for each dataset.

Dataset	Numerical Attribute	Categorical Attribute	Image
MFEAT	Fourier coefficient_1 0.13839	-	
DVM	Length 4865.0	Fuel_type 9	
SUNAttribute	-	Warm 1	
CelebA	-	Big_Nose 0	
PetFinder-adoption	Fee 100	Type 0	
PetFinder-pawpularity	-	Focus 0	
Avito	Price 1290	Category_name 4	

- **MFH [16]:** For MFH method, we set modality general decisive information according to the feature ranking algorithm. The number of the features is fifty percent of that for all features.
- **MMCL [3]:** The same parameters are set for MMCL method according to [3]. We use the frozen version after pretrain and only train the classifier for downstream task.
- **CHARMS:** For FT-Transformer, the number of Transformer blocks is set to 2. We use the K-Means method to cluster the representations obtained by ResNet50 and $n\_cluster$ is 40. Embedding dimension E is set according to the data distribution. Adam optimizer with weight decay is used to train the models. We choose to update cost matrix every 5 epochs, striking a balance between updating them without stable knowledge and minimizing the computational burden. However, we continuously update ϕ throughout the training process to enhance the representation.

Table 2: Comparisons with baseline methods on DVM, SUN, CelebA, Adoption, Pawpularity, and Avito datasets on five random seeds.

	DVM ↑	SUN ↑	CelebA ↑	Adoption ↑	Pawpularity ↓	Avito ↓
LGB	0.9748±0.0014	0.8501±0.0003	0.7963±0.0005	0.4101±0.0053	20.0720±0.0072	0.2290±0.0011
RTDL	0.9682±0.0018	0.8563±0.0011	0.7936±0.0004	0.4107±0.0048	20.0844±0.0098	0.2317±0.0034
ResNet	0.8743±0.0183	0.8361±0.0144	0.8146±0.0092	0.3477±0.0048	18.6150±1.4559	0.2512±0.0034
KD	0.8390±0.0076	0.8382±0.0063	0.8118±0.0046	0.3532±0.0035	19.0683±1.7642	0.2499±0.0015
MFH	–	0.8312±0.0022	0.7507±0.0034	0.3401±0.0027	43.1455±2.0843	0.2873±0.0047
FMR	0.8427±0.0151	0.8347±0.0119	0.8003±0.0143	0.3526±0.0088	19.3517±1.5837	0.2937±0.0084
MMCL	0.8203±0.0040	0.8431±0.0012	0.8041±0.0017	0.2981±0.0026	–	–
CHARMS	**0.9175±0.0052**	**0.8661±0.0032**	**0.8220±0.0022**	**0.3603±0.0037**	**18.4314±0.7427**	**0.2495±0.0025**

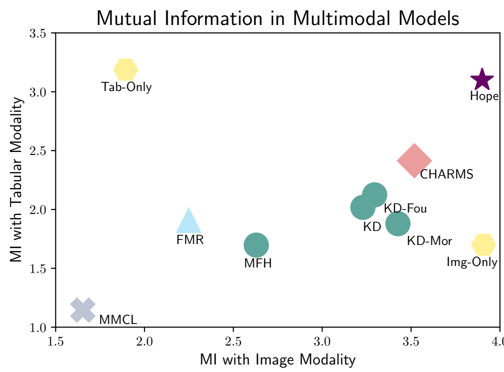


Figure 1: Mutual Information with Different Modality in Multimodal Models. A good model should be able to effectively combine both image and tabular information, resulting in higher mutual information between the two modalities.

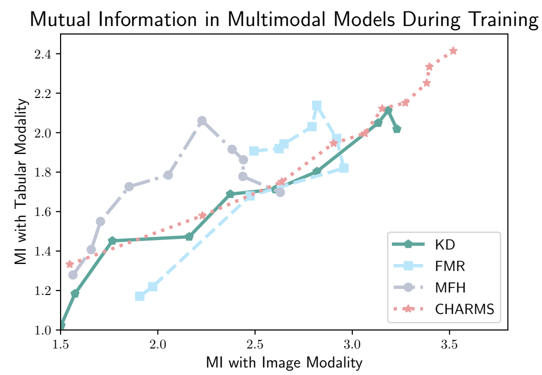


Figure 2: Mutual Information During Training on MVFEAT dataset. We calculate mutual information from the beginning to the convergence process in order to better understand the training process of each method.

We experiment on five random seeds and the results in the form of mean plus standard deviation are shown in the table 2.

A.3 Figure Details

We explain some figures in detail.

- For Figure 4, we calculated the amount of information contained in different modality data for different methods with the MINE method [1]. The image data are simple handwritten digits, we process them simply using a two-layer convolutional neural network, followed by a max pooling layer, and a Dropout layer to prevent overfitting. When calculating the mutual information, we use the $mine$ method as the loss function for approximating the mutual information. The network we choose is a three layer MLP with two hidden layers of size 100, the method we choose is $concat$, and the $batch\_size$ is 16.
- For Figure 2, we do not calculate the mutual information change process for the MMCL method because the MMCL method already performs much less well in Figure 4 than the other baseline models. We hypothesize that MMCL maps the tabular and image representations to another space and therefore the mutual information is lower.
- In the ablation study for different nets, we experimentally validated the impact of different neural network as backbone models on our approach. The accuracy in ORIGIN is {34.77, 34.05, 34.49, 33.98}. The accuracy in out CHARMS is {35.74, 35.52, 35.82, 35.45}.

A.4 Task Details

The usage of knowledge from table to images could be explained from three aspects:

In our setting, the goal is to transfer knowledge from the tabular data to the image model. Both classification and regression tasks are vital and commonly encountered in our setting, where both of them are investigated in our experiments. For instance, on the Adoption dataset, the pet type and size attributes are crucial for the adoption time classification. Guidance on these features in an image would lead to better learning of the image model. Similarly, on the Pawpularity dataset, the eyes and face attributes have a positive assignment on the regression of the popularity of the pet. Therefore, it makes sense to do knowledge transfer from tabular data to image for both classification and regression tasks.

CHARMS is a general method for both classification and regression tasks, in detail, we use cross entropy loss for classification task and mean square error loss for regression task. We achieved an improved image representation by employing the CHARMS method, which leverages the guidance of tabular data on the image data. Specifically, for the classification task, our approach facilitated the representation with a more discerning distribution over the target categories. On the other hand, the regression task enabled us to learn an image representation that better approximated the target values during prediction. The fact that our method performs well on both tasks underscores its generalizability and effectiveness.

Additionally, our visualization experiments provide further evidence of the effectiveness of our method. These experiments reveal that the attributes and channels selected by our approach are appropriately matched, leading to an enhancement in the performance of the image model. This alignment between the attributes and channels serves as strong evidence that we have successfully transferred the relevant knowledge from the table to the image model.

In summary, our approach demonstrates its versatility by excelling in both classification and regression tasks, showcasing its ability to enhance image representations using guidance from tabular data.

B Analysis on Our CHARMS Method

B.1 Comparison with attention method

Our method employs the transfer matrix obtained by OT to weigh the images, with the weights of the corresponding channels raised to learn the tabular attributes. An alternative approach is to use the attention method to weigh the image channels differently and learn each tabular attribute separately, which is a more intuitive approach:

$$\phi(\boldsymbol{x}^T)_{att} = \mathcal{T}(\phi(\boldsymbol{x}^T)) \cdot \phi(\boldsymbol{x}^T) \tag{1}$$

where $\mathcal{T}$ is a two layer MLP that first downscales the image representation obtained by ϕ before rescaling it to its original dimension, thereby weighting the different channels of the image.

In contrast to our method CHARMS, this method assigns a weight to each input element so that the model can pay more attention to those input elements that are more important for the task at hand. The attention method constructs a learnable mask for each attribute and learns each attribute separately based on the backbone network. However, this approach may result in unequal impacts of different masks on the main task. In contrast, our method weights the attention of different channels in the representation obtained by the main task, which essentially corrects the main task while avoiding potential inconsistency issues caused by the attention method.

We compare the performance of our method CHARMS with the attention method in all experiments and summarized the results in Table 4. The table shows that the attention method did not perform as well as our method on all datasets. Specifically, on the DVM dataset, which involves a complex downstream task of 129 classification tasks, the attention method constructed different attentions for different attributes, which confused the backbone network and led to a decrease in overall task performance.

This finding highlights the impracticality of using the attention mechanism alone to integrate the abundant information in tabular data into the image model. This further supports the effectiveness of our proposed approach.

Table 3: Comparison with CLIP method. Here CLIP-LP means two encoders are fixed, and only the classification head is trained. CLIP-FT means fine-tuning the entire CLIP network.

	DVM↑	SUN ↑	CelebA ↑	Adoption ↑	Pawpularity ↓	Avito ↓
CLIP-LP	0.7619	0.6918	0.7590	0.3047	20.1537	0.2972
CLIP-FT	0.8417	0.8333	0.8165	0.2935	42.9489	0.2940
CHARMS	**0.9175**	**0.8661**	**0.8220**	**0.3603**	**18.4314**	**0.2495**

Table 4: Comparison with Attention method. Here Attention means we directly conduct the attention mechanism on the feature extracted by ϕ and learn an attention mask for all tabular attributes.

	DVM ↑	SUN ↑	CelebA ↑	Adoption ↑	Pawpularity ↓	Avito ↓
Attention	0.4757	0.8550	0.8180	0.3454	18.7401	0.2544
CHARMS	**0.9175**	**0.8661**	**0.8220**	**0.3603**	**18.4314**	**0.2495**

B.2 Comparison with CLIP method

CLIP is pre-trained on a large amount of text and image pairs, which makes it able to map from text to images. Some previous studies have demonstrated that CLIP is able to transform tabular data to text for classification given the column names [15, 4]. However, CLIP is heavily reliant on the semantic information contained within the text, so that the semantics of attributes are inevitable.

Indeed, the setting of this paper is more general. We expect to transfer the tabular knowledge to the image modality during training to cope with the absence of expert knowledge during testing. Our method CHARMS aims to automatically extract the semantic information from the tabular and align it with the corresponding image channels without requiring explicit knowledge of the attribute's precise meaning. Specifically, as we show in Section 4.2, based on measuring the similarity across attributes and channels, OT discovers and aligns the attribute semantic automatically.

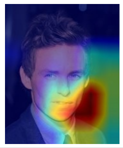
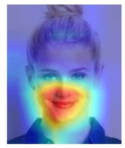
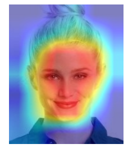
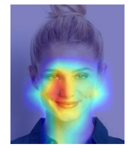
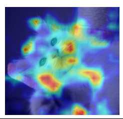
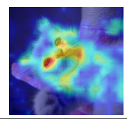
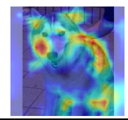
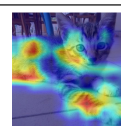
We conducted an experiment with CLIP. In this experiment, we converted the tabular data into text format, such as "length: 16". To ensure a fair comparison, we utilized CLIP from **??** with the ResNet50 backbone. The CLIP model consists of an image encoder and a textual encoder, and we connected a one-layer linear head for classification or regression after the image encoder. Two versions of CLIP were trained in our experiment. CLIP-LP means CLIP-LinearProb, which denotes the scenario where the two encoders are fixed, and only the classification head is trained. CLIP-FT means CLIP-FineTune, on the other hand, involves fine-tuning the entire CLIP network. With the contrastive learning of the two modalities of the CLIP model, tabular knowledge is transferred to the image modality. By transforming the task into a language-to-vision knowledge transfer, the results were obtained in table 3.

From the experiments, we can see that the performance of CLIP is not ideal. This is probably due to the fact that in tabular data, each column holds its own distinct meaning, and directly utilizing it as input to CLIP can lead to the loss of certain information. For instance, on the CelebA dataset, the attribute "wood (not part of a tree)" might not be a highly significant feature. However, when this attribute is converted to text format, its character length tends to be relatively long, which can introduce redundancy in the information.

From another perspective, previous work has pointed out that there is a modality gap in the CLIP's embedding space [7]. This gap is caused by a combination of model initialization and contrastive learning optimization. In a multi-modal model with two encoders, the representations of the two modalities are clearly apart when the model is initialized. During optimization, contrastive learning keeps the different modalities separate by a certain distance. This gap makes the CLIP method fail in our task.

In summary, the loss of information and the modality gap that arises when transferring tabular data to images can hinder the performance of the CLIP method in our setting. However, our method addresses these challenges by automatically discovering and establishing the matching relationship between the two modalities, thereby facilitating effective knowledge transfer, which is a more general method.

Table 5: More Visualization by GradCAM.

Tabular Attribute	5_o_Clock_Shadow	Arched_Eyebrows	Big_Nose	Blond_Hair
Aligned Channel	65, 87, 119, 236...	33, 76, 78, 115, ...	50, 224, 258, ...	684
Visualization				
Tabular Attribute	High_Cheekbones	Smiling	Oval_Face	Rosy_Cheeks
Aligned Channel	2, 26, 41, 85,...	11, 12, 28, 57, ...	52, 646, 924, ...	4, 47, 88,...
Visualization				
Tabular Attribute	Type		Color	
Aligned Channel	399, 413, 414, 521...		400, 412, 425, 448...	
Visualization				
Aligned Channel	399, 413, 414, 521...		400, 412, 425, 448...	
Visualization				

C More Experiments

C.1 More Visualization

We provide more visualizations in Table 5 to validate the ability of CHARMS to match the corresponding attributes and channels. We apply GradCAM on various datasets, which show similar visualization results, where the channels could be matched to a certain attribute with semantic meaning.

For the Adoption dataset, all tabular attributes are inherently more abstract in nature. However, for the purpose of visualization, we have specifically selected features that are visually recognizable by humans from images. For instance, attributes such as the type of pet and the color of the pet highlight more general aspects that are of interest.

From the visualization, we can see that the judgment of the pet type focuses more on the pet's head, whereas the judgment of the color takes into account the whole body of the pet, and from this point of view we believe that our approach does achieve knowledge transfer.

C.2 Visualization with t-SNE

To visualize the impact of our method on the distribution of image features, we conducted experiments using the t-SNE method [14]. t-SNE can map high-dimensional data to a two- or three-dimensional space, enabling better visualization and interpretation of the data structure. The method employs a nonlinear mapping approach that minimizes the difference between the distances of points in high-dimensional space and those in low-dimensional space. Specifically, it represents high-dimensional data points as probability distributions and generates corresponding probability distributions in the low-dimensional space. Then, it uses KL divergence to measure the difference between the two probability distributions and minimizes it to achieve the best mapping effect.

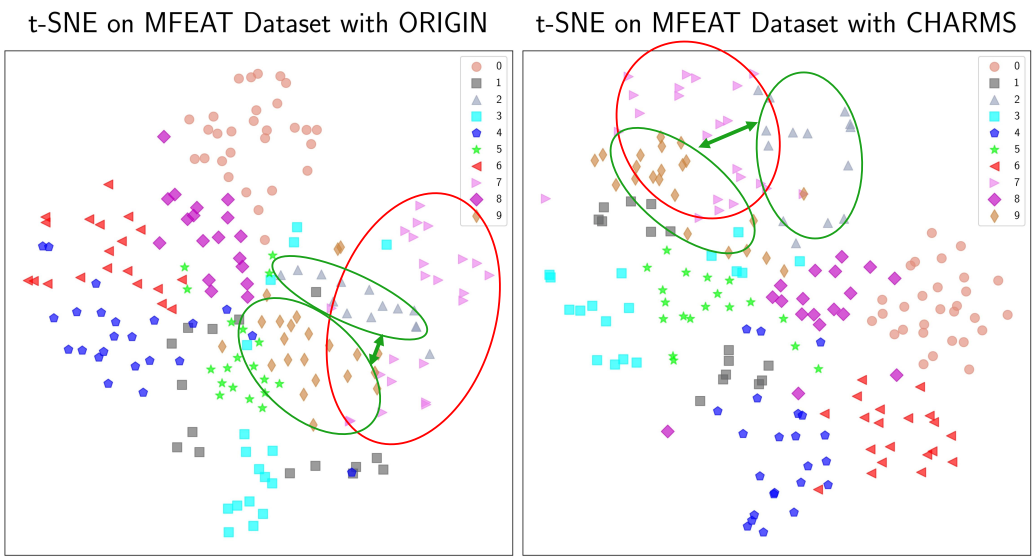


Figure 3: Visualization of t-SNE on the MFEAT dataset. the ORIGIN method represents training on image modalities only. As can be seen from the figure, our method makes the intra-class distance smaller and the inter-class distance larger. Therefore the transfer of expert knowledge from tabular data to the image model is effective. The red circles mean that our method makes the intra-class distance smaller, and the green circles indicate that our method makes the inter-class distance larger.

The experimental results are presented in Figure 3, where the ORIGIN method refers to training with image modalities only. The figure shows that the ORIGIN method achieved good segmentation results due to the task's simplicity. However, due to the lack of expert knowledge, the intra-class distance is still large, particularly for samples with label 7, while the inter-class distances remain small, such as for samples with labels 2 and 9. In contrast, our method compensates for these deficiencies by transferring expert knowledge.

C.3 More Mutual Information experiments

We chose the MFEAT dataset for the Mutual Information experiments since, in this dataset, the formal features of each category are simple and easily distinguishable. For example, morphological features and non-morphological features. And the images are all digital images, which are relatively simple and easy to understand. The experiment mainly helps us understand. More mutual information experiments can be obtained in Table 4 5.

The experiments in PetFinder-adoption dataset also indicate that existing methods for transferring tabular knowledge to image models yield low mutual information between the representations and tabular data. Our CHARMS method, on the other hand, maximises the mutual information of tabular and images to achieve better results.

C.4 More Ablation Studies

In the CHARMS method, we use the K-Means [9, 10] method to cluster the 2048-dimensional features extracted from ResNet. We discuss the number of clusters on the SUNAttribute dataset, and the results in Table 6 show that the performance of CHARMS is not affected by the number of clusters taken, demonstrating the robustness of the method to hyperparameter choices. This robustness makes the method more flexible and reliable in practical applications, as it does not require excessive hyperparameter tuning or fine-tuning, saving time and effort.

To further demonstrate the applicability and robustness of our proposed method, CHARMS, we conducted experiments using different network structures on DVM dataset with results shown in Table 7. The result also shows that the performance improvements achieved by our method are consistent across different network structures.

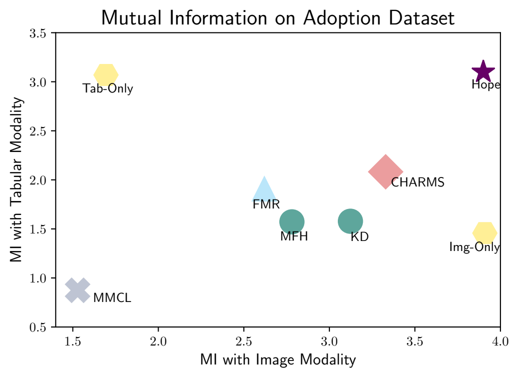


Figure 4: Mutual Information with Different Modality on the Adoption Dataset.

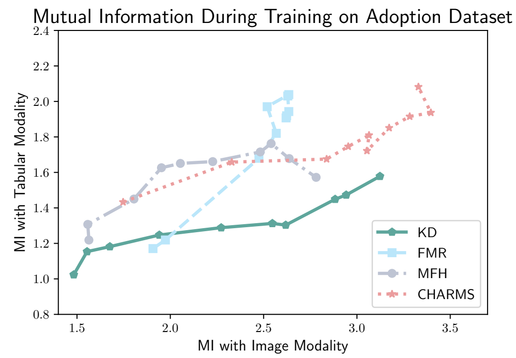


Figure 5: Mutual Information During Training on the Adoption dataset.

Table 6: Ablation study on cluster number on SUNAttribute dataset.

n_cluster	20	40	60	80	100
Accuracy	0.8494	0.8661	0.8494	0.8556	0.8522

D Limitations and Future Works

Our approach relies on leveraging mutual information between the two modalities, which establishes the feasibility of knowledge transfer. When there is a significant amount of mutual information present between the tabular and image modalities, our approach can effectively transfer relevant knowledge and insights between them. On the other hand, converting text into tables is indeed a viable approach, but this approach results in the loss of some of the textual information and it is challenging to handle such a conversion well. The problem of testing data drift also exists in real life. We will consider this problem deeply in future work. In terms of social impact, we think that our approach holds potential for application in the medical field, where it can assist doctors in making rapid and accurate diagnoses. There should be no negative social impact of our method.

Our work demonstrates the effectiveness of our method in both classification and regression tasks. In future work, it would be valuable to investigate the applicability of our method to other tasks, such as semantic segmentation. These types of tasks may require additional domain-specific knowledge, such as precise object localization within images, to achieve optimal performance. Nonetheless, we believe that our approach is still applicable for such tasks.

On the other hand, the high cost of annotating expert data often leads to imbalanced datasets, which pose a challenge for improving image model performance using a limited amount of tabular data. Therefore, addressing this data imbalance is crucial for future work.

References

[1] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.

Table 7: Impact of different network structures on the method on DVM dataset.

	ResNet	DenseNet	Inception	MobileNet
Model Size / M	25.8	8.2	6.8	3.7
ORIGIN	0.8743	0.8671	0.7492	0.8206
CHARMS	0.9175	0.9115	0.9012	0.8961

[2] Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. *Advances in Neural Information Processing Systems*, 34:18932–18943, 2021.

[3] Paul Hager, Martin J Menten, and Daniel Rueckert. Best of both worlds: Multimodal contrastive learning with tabular and imaging data. *arXiv preprint arXiv:2303.14080*, 2023.

[4] Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. Tabllm: Few-shot classification of tabular data with large language models. In *International Conference on Artificial Intelligence and Statistics*, pages 5549–5581, 2023.

[5] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[6] Jingmin Huang, Bowei Chen, Lan Luo, Shigang Yue, and Iadh Ounis. Dvm-car: A large-scale automotive dataset for visual marketing research and applications. In *2022 IEEE International Conference on Big Data*, pages 4140–4147, 2022.

[7] Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625, 2022.

[8] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision*, pages 3730–3738, 2015.

[9] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28:129–137, 1982.

[10] J MacQueen. Classification and analysis of multivariate observations. In *5th Berkeley Symp. Math. Statist. Probability*, pages 281–297, 1967.

[11] Genevieve Patterson, Chen Xu, Hang Su, and James Hays. The sun attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision*, 108:59–81, 2014.

[12] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015.

[13] Martijn van Breukelen, Robert PW Duin, David MJ Tax, and JE Den Hartog. Handwritten digit recognition by combined classifiers. *Kybernetika*, 34:381–386, 1998.

[14] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[15] Zifeng Wang and Jimeng Sun. Transtab: Learning transferable tabular transformers across tables. *arXiv preprint arXiv:2205.09328*, 2022.

[16] Zihui Xue, Zhengqi Gao, Sucheng Ren, and Hang Zhao. The modality focusing hypothesis: Towards understanding crossmodal knowledge distillation. *arXiv preprint arXiv:2206.06487*, 2022.

[17] Yang Yang, De-Chuan Zhan, Ying Fan, Yuan Jiang, and Zhi-Hua Zhou. Deep learning for fixed model reuse. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2831–2837, 2017.