



Transpose and Mask: Simple and Effective Logit-Based Knowledge Distillation for Multi-attribute and Multi-label Classification

Yuwei Zhao, Annan Li^(✉), Guozhen Peng, and Yunhong Wang

State Key Laboratory of Virtual Reality Technology and Systems,
School of Computer Science and Engineering, Beihang University,
Beijing 100191, China

{yuweizhao, liannan, guozhen_peng, yhwang}@buaa.edu.cn

Abstract. Knowledge distillation (KD) improves a student network by transferring knowledge from a teacher network. Although KD has been extensively studied in single-labeled image classification, it is not well explored under the scope of multi-attribute and multi-label classification. We observe that the logit-based KD method for the single-label scene utilizes information from multiple classes in a single sample, but we find such logits are less informative in the multi-label scene. To address this challenge in the multi-label scene, we design a *Transpose* method to extract information from multiple samples in a batch instead of a single sample. We further note that certain classes may lack positive samples in a batch, which can negatively impact the training process. To address this issue, we design another strategy, the *Mask*, to prevent the influence of negative samples. To conclude, we propose **Transpose and Mask Knowledge Distillation (TM-KD)**, a simple and effective logit-based KD framework for multi-attribute and multi-label classification. The effectiveness of TM-KD is confirmed by experiments on multiple tasks and datasets, including pedestrian attribute recognition (PETA, PETA-zs, PA100k), clothing attribute recognition (Clothing Attributes Dataset), and multi-label classification (MS COCO), showing impressive and consistent performance gains.

Keywords: Knowledge distillation · multi-attribute classification · multi-label classification

1 Introduction

Recent deep learning research has revealed that increasing the model capacity properly often leads to improved performance [6, 7, 10]. Nevertheless, the larger model usually comes with potential drawbacks, such as long training time, increased inference latency, and high GPU memory consumption.

To address this issue, knowledge distillation (KD) [11, 21] is often employed, which utilizes a strong teacher network to transfer knowledge to a relatively

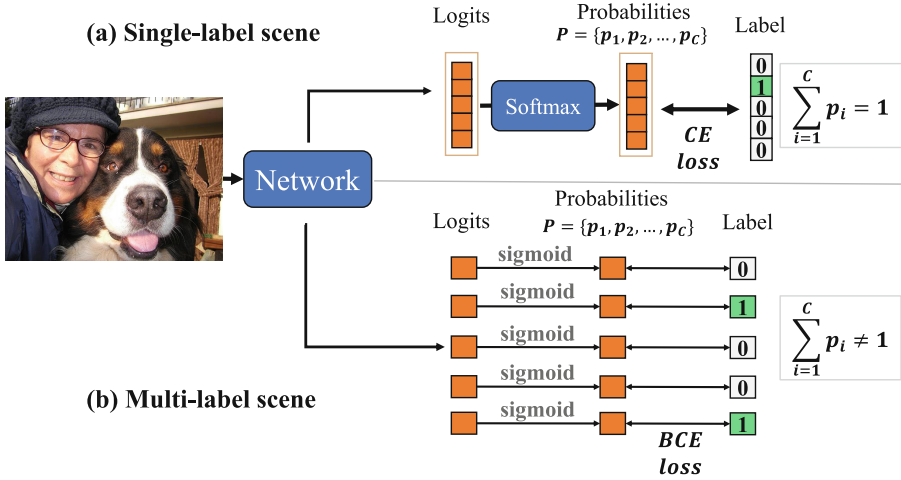


Fig. 1. The different training objectives in the single-label scene and the multi-label scene. Due to the property of the softmax function, logits in the single-label scene sum up to 1, while logits in the multi-label scene don't have such property.

weaker student network. KD has already been extensively studied in the single-labeled image classification task, where each sample has only one label. However, it has not been well investigated in the multi-attribute and multi-label classification (we use the multi-label scene to refer to them in this paper), where each sample may have multiple labels.

In this paper, we focus on exploring logit-based Knowledge Distillation (KD) in the context of the multi-label scene. The logit-based KD method stands out due to its simplicity in both idea and implementation, its independence from the backbone model structure, as well as its relatively low computational overhead compared to other KD methods [9]. In this method, the term *logits* refers to the outputs of the neural network's final layer, which will then be fed into a softmax function.

However, for the following reasons, we do not directly employ the logit-based KD method [11] (which we call vanilla KD below) commonly used in single-label image classification.

As shown in Fig. 1, in the single-label scene, a softmax function is applied to the logits to generate predictions in terms of probabilities. As the sum of prediction for all classes equal 1, the logits of different classes in a single sample become highly correlated, and therefore, contain interdependent information. This entanglement of logits has also been noted by Decoupled KD [28]. However, in the context of the multi-label scene, the logits of different classes are used to calculate loss with their own labels and do not explicitly interact with the logits of other classes. Such lack of interaction between logits of different classes weaken the information contained in the relation of logits of different classes. Since the vanilla KD distills exactly the relation of logits of different classes within a sample, directly employing the vanilla KD in the multi-label scene leads to a reduced amount of information.

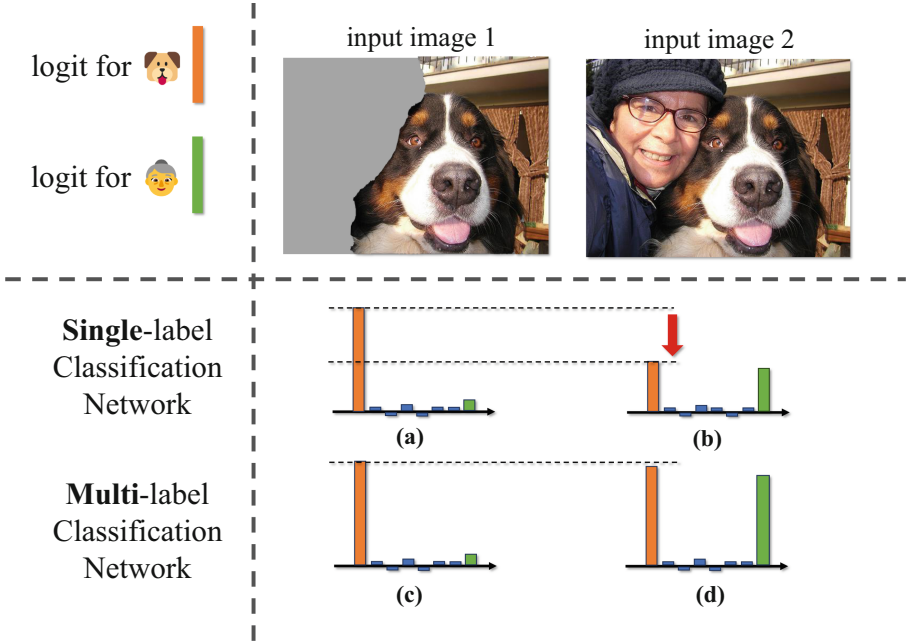


Fig. 2. In the single-label scene, the logit of a same dog can be affected by the presence of the person, as shown in (a) and (b). Instead, in the multi-label scene, the logit of a same dog tends to be more stable and less influenced by the other classes, as shown in (c) and (d).

On the contrary, when evaluating logits from multiple samples, logits of the same class across multiple samples in the multi-label scene are more comparable than those in the single-label scene. To demonstrate this idea, we present an example in Fig. 2, where two input images contain the exactly same dog, with the only difference being a person in the second image. In a single-label scene, the softmax function results in $\sum_{i=1}^C p_i = 1$, which makes the rise in person lead to the decrease in the dog. However, in the multi-label scene, such a decrease is not as significant because the logit of the person does not explicitly influence the logit of the dog. To conclude, if we look into the logits of the same class in different samples, their values are not comparable in the single-label scene but are comparable in the multi-label scene.

Inspired by the aforementioned two observations, we propose to distill knowledge in logits from the same class and different samples, rather than from the same sample and different classes. We refer to this strategy as the *Transpose*.

We also note that the multi-label scene typically exhibits a class imbalance, where most classes or attributes contain fewer (or significantly fewer) positive samples compared to negative samples. Therefore, under our transpose strategy, there are more negative samples than vanilla KD [11]. We assume that the relative relation in negative samples is less informative, since from two positive

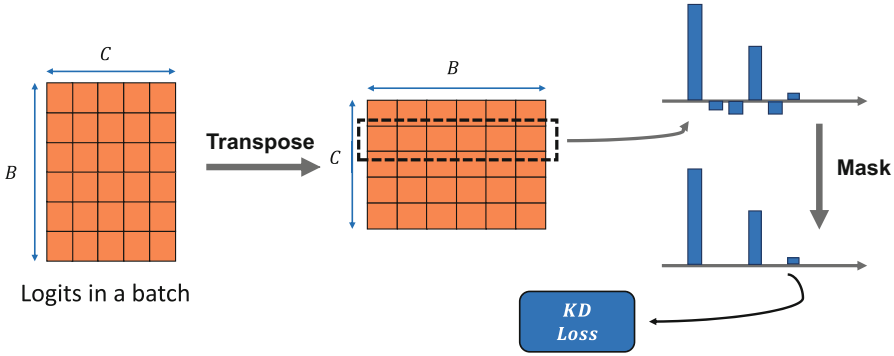


Fig. 3. Illustration of our proposed Transpose and Mask Knowledge Distillation (TM-KD), where logits are first transposed and are then masked with 0 according to negative samples predicted by the teacher.

samples the network can learn from salient information (for example, the cat in the i -th image is more salient than the j -th image) but such information does not exist in two negative samples (since they simply contain no cat). We thus propose another strategy to fill all position whose logits in the teacher network is negative (negative samples predicted by the teacher) with zero before distillation, which we refer to the *Mask* strategy.

As illustrated in Fig. 3, based on the above analysis, we propose **Transpose and Mask Knowledge Distillation (TM-KD)**, which is a simple but effective logit-based knowledge distillation method. We further validate the effectiveness of our method on three tasks and five datasets, which shows TM-KD is better than both the vanilla student network and the student network with vanilla logit-based KD.

2 Related Work

Knowledge Distillation. Knowledge distillation (KD), proposed by Hinton et al. [11], aims to utilize a strong teacher network for a better student network. KD methods can be roughly divided into logit-based [11, 13, 15, 26] and feature-based [2, 21]. KD is originally proposed for single-label image classification but recent studies also show the effectiveness of KD in other tasks like object detection [4, 29], semantic segmentation [19, 24], graph neural network [8, 25], anomaly detection [3] and some low-level tasks [23].

Multi-attribute and Multi-label Learning. Due to the limited space, it is hard to provide a detailed overview of each task. Here we provide some representative methods for these tasks. Label2Label [14] proposes a language modeling framework for clothing attribute recognition and pedestrian attribute recognition. JLAC [22] exploits graph neural network on top of convolution neural

network for better results for pedestrian attribute recognition. Query2Label [17] proposes a simple transformer for better modelling multi-label classification. Note that our work doesn't focus on the state-of-the-art performance of each dataset and thus is orthogonal to these works.

KD for the Multi-label Scene. So far, KD for the multi-label scene is not well explored. Liu et al. [20] leverages the extra information from weakly-supervised detection for KD in the multi-label scene. Zhang et al. [27] proposes a feature-based method for KD in the multi-label scene exploiting class activation maps. On the contrary, our work focuses on better logit-based KD and uses no auxiliary model like the object detector.

3 Method

3.1 Preliminaries

A training batch with B samples and C classes for the multi-label scene can be described as $D = \{(x_i, y_i), i = 1, 2, \dots, B\}$, where x_i is the i -th image in a batch and $y_i \in \{0, 1\}^C$ is a binary vector with length C , the labels for i -th sample. We used y_{ij} to represent the j -th attribute label for i -th sample and $y_{ij} = 1$ for a positive sample, $y_{ij} = 0$ for a negative sample.

Then, a classification network f is trained to predict a vector $z_i \in \mathbb{R}^C$ for the i -th sample and $z \in \mathbb{R}^{B \times C}$ is called **logits**. In the multi-label scene, each separate logit is then fed into *sigmoid* function, and then calculate the binary cross-entropy (BCE) loss. The above process can be formally defined as:

$$z_i = f(x_i) \quad (1)$$

$$p_{ij} = \frac{1}{(1 + e^{-z_{ij}})} \quad (2)$$

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{BC} \sum_{i=1}^B \sum_{j=1}^C y_{ij} \log(p_{ij}) + (1 - y_{ij}) \log(1 - p_{ij}) \quad (3)$$

And if KD is applied during the training, the final loss can be represented as:

$$\mathcal{L} = \mathcal{L}_{\text{BCE}} + \lambda \mathcal{L}_{\text{KD}} \quad (4)$$

where λ is a hyperparameter to balance BCE loss and KD loss.

Below we will show the different designs of \mathcal{L}_{KD} in vanilla KD and our TM-KD.

3.2 Vanilla KD

Apart from the student network f^s , KD uses a stronger teacher network f^t trained beforehand to help the student network. The logits of them can be represented as $z^s \in \mathbb{R}^{B \times C}$ and $z^t \in \mathbb{R}^{B \times C}$.

Directly using vanilla KD from the single-label classification, Kullback-Leibler (KL) divergence loss is used to minimize the discrepancy of probabilities from different classes in the same sample, which is:

$$p_i^t = \text{softmax}(z_i^t/\tau), \quad p_i^s = \text{softmax}(z_i^s/\tau), \quad i = 1, 2, \dots, B \quad (5)$$

$$\mathcal{L}_{\text{KD}} = \frac{1}{B} \sum_{i=1}^B KL(p_i^t, p_i^s) \quad (6)$$

where τ is a hyperparameter to adjust the smoothness of two probabilistic distributions.

Algorithm 1: PyTorch-style pseudocode for vanilla KD and TM-KD.

```

T = 1
KLdiv = nn.KLDivLoss(reduction="batchmean")
def KD_from_logit(logit_stu, logit_tea):
    log_prob = F.log_softmax(logits_stu/T, dim=1)
    prob = F.softmax(logits_tea/T, dim=1)
    loss = KLdiv(log_prob, prob)
    return loss

def vanillaKD(logit_stu, logit_tea):
    # logit_stu.shape : [B, C]
    return KD_from_logit(logit_stu, logit_tea)

def TM_KD(logit_stu, logit_tea):
    # logit_stu.shape : [B, C]
    mask = (logit_tea[0]<0).int()
    logits_stu = logits_stu.masked_fill(mask, 0)
    logits_tea = logits_tea.masked_fill(mask, 0)
    logits_stu = logits_stu.permute(1,0)
    logits_tea = logits_tea.permute(1,0)
    # logit_stu.shape : [C, B]
    return KD_from_logit(logit_stu, logit_tea)

```

3.3 TM-KD

As illustrated in Fig. 3, our TM-KD consist of two strategies, i.e. the Transpose and the Mask respectively.

For the Mask, to alleviate the influence of useless information in negative samples for a class, we set the position to zero if the corresponding logits in the teacher network are negative. By doing so, the teacher network only distills the knowledge in positive samples from its perspective. Formally:

$$\hat{z}_{ij}^* = \begin{cases} z_{ij}^*, & z_{ij}^t \geq 0 \\ 0, & z_{ij}^t \leq 0 \end{cases} \quad (7)$$

where $* \in \{s, t\}$. Note the student network and teacher network share the same mask from the teacher network.

For the Transpose, we no longer distill from the different classes in the same sample, but from the different samples in the same class, which is done by:

$$p_j^t = \text{softmax}(\hat{z}_{*j}^t/\tau), \quad p_j^s = \text{softmax}(\hat{z}_{*j}^s/\tau), \quad j = 1, 2, \dots, C \quad (8)$$

$*$ can be any i where $1 \leq i \leq B$, and $\hat{z}_{*j}^t, \hat{z}_{*j}^s \in \mathbb{R}^B$ can be viewed as logits of the same class in different samples. We have analyzed why they contain more information in the multi-label scene in the introduction.

We also provide a pseudocode of vanilla KD and TM-KD in Algo. 1.

4 Experiments

In this section, we validate the performance of our TM-KD on three tasks and five datasets. Our TM-KD consistently demonstrates impressive performance across all the datasets. In addition, we conduct ablation studies to demonstrate the effectiveness of our Transpose and Mask strategy.

4.1 Experimental Setting

We conduct our experiment under two KD settings with ResNet [10], where ResNet-101 serves as the teacher model for teaching ResNet-50, and ResNet-50 serves as the teacher model for teaching ResNet-18. We train a teacher first and then utilize it to help train a student. The only exception is that we used ResNet-101 as the teacher and ResNet-34 as the student for the MS COCO dataset. Our code is on top of the codebase by Jia et al. [12] and the teacher network is retrained instead of loaded. Below, we'll present more experimental details.

Datasets. We evaluate on pedestrian attribute recognition using PETA [5], PETA-zs [5, 12], and PA100k [18], clothing attribute recognition using the Clothing Attributes Dataset [1], and multi-label classification using MS COCO [16]. We list the statistics of used datasets in Table 1. We utilize default dataset split and more details can be found in their original paper and codebase by Jia et al. [12]. Note that for the clothing attributes attribute dataset, we use only 22 out of 26 attributes, where we exclude attributes of *sleeve length*, *neckline*, *category* and *gender*.

Implementation Details. For KD hyperparameters, we set $\lambda = 20$ in Eq. 4 and set $\tau = 1$ in Eq. 8 for all experiments. It turns out that the order of magnitude of λ and τ (in the power of 10) does affect the results, but when λ and τ are in the same order of magnitude, the exact values of them don't affect the result. Since we implemented our method on top of the codebase by Jia et al. [12], we used most of the default settings from it. It's guaranteed that hyperparameters for different methods on the same dataset are also the same.

Table 1. Statistics of 5 used datasets from 3 different tasks. **PAR**: pedestrian attribute recognition. **CAR**: clothing attribute recognition. **MLC**: multi-label classification. N_{train} : number of samples in train set. N_{test} : number of samples in test set. N_{attr} : number of attributes used in this dataset.

Task	Dataset	N_{train}	N_{test}	N_{attr}
PAR	PETA	11400	7600	35
	PETA-zs	15067	3933	35
	PA100k	90000	10000	26
CAR	Clothing Attributes	1500	356	22
MLC	MS COCO	82081	40137	80

Metrics. Following the routine in previous works, We report mean accuracy (mA) and $micro-F1$ for pedestrian attribute recognition and clothing attribute recognition datasets and report mA for the multi-label classification dataset. Since ReduceLROnPlateau learning rate scheduler is used following the codebase by Jia et al. [12], we report the metrics after the first epoch of learning rate reducing to 10^{-5} for pedestrian attribute recognition datasets, and we report those of clothing attribute recognition dataset for first reduction to 10^{-6} . For the multi-label classification dataset, we report the metrics at the last (30) epoch.

Table 2. The results for pedestrian attribute recognition, Δ_* represent the performance gains from our TM-KD compared with the baseline w/o KD and the baseline with vanilla KD. The rightmost column reports the average of 3 datasets.

KD Setting	KD Method	PETA		PETA-zs		PA100k		Avg.	
		mA	F1	mA	F1	mA	F1	mA	F1
ResNet50 ↓ ResNet18	Teacher	84.21	86.21	71.07	72.12	80.27	87.32	78.51	81.88
	w/o KD[12]	81.76	84.85	70.70	71.54	77.97	85.92	76.81	80.77
	Vanilla KD[11]	81.95	85.18	69.29	71.70	78.54	86.33	76.59	81.07
	TM-KD(Ours)	84.75	85.02	73.54	71.77	81.66	85.96	79.98	80.91
	$\Delta_{w/o}$	+2.99	+0.17	+2.84	+0.23	+3.69	+0.04	+3.17	+0.14
	$\Delta_{vanilla}$	+2.80	-0.16	+4.25	+0.07	+3.12	-0.37	+3.39	-0.16
ResNet101 ↓ ResNet50	Teacher	85.13	87.11	71.84	72.25	80.66	87.59	79.21	82.31
	w/o KD[12]	84.21	86.21	71.07	72.12	80.27	87.32	78.51	81.88
	Vanilla KD[11]	84.04	86.35	71.07	72.76	80.22	87.34	78.44	82.15
	TM-KD(Ours)	86.72	85.83	73.40	72.62	82.74	87.29	80.95	81.91
	$\Delta_{w/o}$	+2.51	-0.38	+2.33	+0.50	+2.47	-0.03	+2.44	+0.03
	$\Delta_{vanilla}$	+2.68	-0.52	+2.33	-0.14	+2.52	-0.05	+2.51	-0.24

4.2 Main Results

Pedestrian Attribute Recognition. We report our result in Table 2. It can be seen that vanilla KD has negligible influence on baseline. And we compared TM-KD with the baseline w/o KD and with vanilla KD in Δ_* rows. Our TM-KD has impressive and consistent gains on all datasets w.r.t mA .

When it comes to $F1$, our model isn't as outstanding as its performance w.r.t mA but still gets an overall positive delta on the average performance compared to the baseline w/o KD. We argue that mA is the main metric for pedestrian attribute recognition since it calculates the mean accuracy over classes, while *micro* - $F1$ treats all samples equally and thus can't well reflect the model's performance in class imbalance scene. Jia et al. [12] also note that the trade-off exists between mA and $F1$, and they show that by changing the weight function we can control the trade-off lean to mA or $F1$ to some extents.

Table 3. Our results on the Clothing Attributes Dataset. Note that our TM-KD has even helped student's performance surpass their corresponding teacher.

KD Setting	Teacher	w/o KD	Vanilla KD	TM-KD(Ours)
	(mA/F1)	(mA/F1)	(mA/F1)	(mA/F1)
R50→R18	65.2/48.7	62.3/41.0	61.1/39.5	68.7/50.2
R101→R50	66.4/50.0	65.2/48.7	64.5/47.4	71.3/55.2

Clothing Attribute Recognition. Our results on clothing attribute recognition are presented in Table 3, wherein our TM-KD demonstrates more remarkable performance. Surprisingly, our ResNet-18 student, trained by ResNet-50, outperforms even the ResNet-101 teacher. Additionally, the ResNet-50 taught by ResNet-101 with our TM-KD also achieves significantly better results compared to all other methods. On the contrary, the vanilla KD approach leads to performance degradation for both ResNet-18 and ResNet-50.

One possible reason for such remarkable performance may be the fact that the Clothing Attributes Dataset contains a very limited number of samples (recall Table 1). Intuitively, when the training data is extremely insufficient to train a network, the network has even more potential to progress. Consider two college students, and in their final exams one gets a D^- grade while another gets an A grade. If we teach them in the same way, apparently the former will progress more.

Table 4. Our results on the MS COCO dataset.

ResNet-101	ResNet-34		
	w/o KD [12]	vanilla KD [11]	TM-KD(ours)
83.04	78.81	77.11	80.83

Multi-label Classification. As shown in Table 4, our results also boost the performance of the student ResNet-34 (+2.02) on the MS COCO multi-label dataset, contrary to the negative impact caused by vanilla KD. Although the improvement may not be significant, considering the difficulty of this dataset and performance degradation of the vanilla KD, the result is still quite impressive.

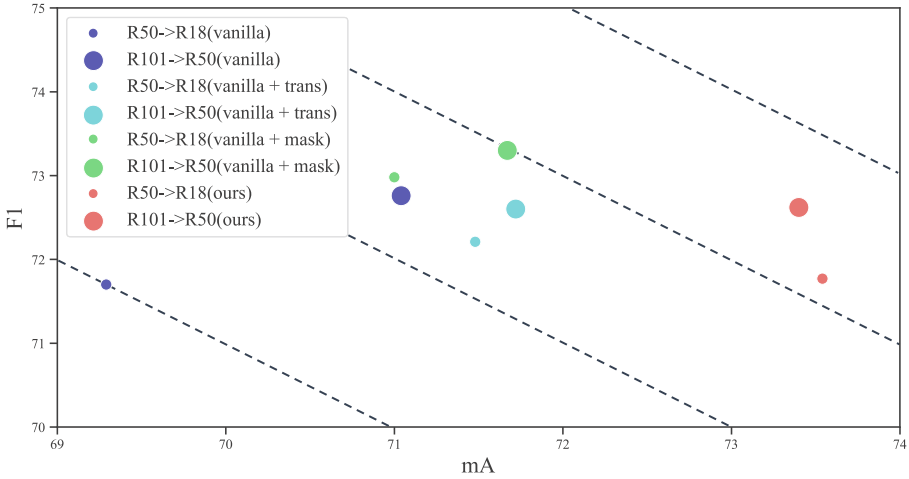


Fig. 4. Ablation of the proposed Transpose strategy and Mask strategy on the PETA-zs dataset. The dark blue dashed line assumes the 1:1 trade-off between mA and F1. (Color figure online)

4.3 Ablation Study

To evaluate the effectiveness of the two proposed strategies, we conduct an ablation study by incorporating one of them into the vanilla KD method. The corresponding results are presented in Fig. 4. As discussed previously in Sect. 4.2, there exists a trade-off between mA and F1 scores in pedestrian attribute recognition. To provide a better visual representation, we assume an equal trade-off ratio of 1:1 and plot a dark blue dashed line. Under this assumption, points on the same line are considered equally effective. It can be seen that applying only one of our strategies can also improve the performance compared with vanilla KD, validating the effectiveness of the proposed two strategies. And when used together in our TM-KD, the performance becomes even better.

5 Conclusion

In this paper, we analyze the logits in the single-label scene and the multi-label scene and then propose TM-KD (**T**ranspose and **M**ask **K**nowledge **D**istillation), a simple and effective logit-base KD method for multi-attribute and multi-label

classification. The proposed method is evaluated on five datasets of three tasks. While vanilla KD usually brings nearly no improvement and sometimes even degradation, TM-KD gets impressive and consistent results on all datasets, validating the effectiveness of TM-KD.

Acknowledgment. This work was supported by National Natural Science Foundation of China under Grant U20B2069.

References

1. Chen, H., Gallagher, A., Girod, B.: Describing clothing by semantic attributes. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7574, pp. 609–623. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33712-3_44
2. Chen, P., Liu, S., Zhao, H., Jia, J.: Distilling knowledge via knowledge review. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5008–5017 (2021)
3. Cheng, H., Yang, L., Liu, Z.: Relation-based knowledge distillation for anomaly detection. In: Ma, H., et al. (eds.) PRCV 2021. LNCS, vol. 13019, pp. 105–116. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-88004-0_9
4. Dai, X., et al.: General instance distillation for object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7842–7851 (2021)
5. Deng, Y., Luo, P., Loy, C.C., Tang, X.: Pedestrian attribute recognition at far distance. In: Proceedings of the 22nd ACM International Conference on Multimedia, pp. 789–792 (2014)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint [arXiv:1810.04805](https://arxiv.org/abs/1810.04805) (2018)
7. Dosovitskiy, A., et al.: An image is worth 16×16 words: transformers for image recognition at scale. In: International Conference on Learning Representations (2021). <https://openreview.net/forum?id=YicbFdNTTy>
8. Feng, K., Li, C., Yuan, Y., Wang, G.: Freekd: free-direction knowledge distillation for graph neural networks. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 357–366 (2022)
9. Gou, J., Yu, B., Maybank, S.J., Tao, D.: Knowledge distillation: a survey. *Int. J. Comput. Vision* **129**, 1789–1819 (2021)
10. He, K., Zhang, X., Ren, S., et al.: Deep residual learning for image recognition. In: CVPR (2016)
11. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint [arXiv:1503.02531](https://arxiv.org/abs/1503.02531) (2015)
12. Jia, J., Huang, H., Chen, X., Huang, K.: Rethinking of pedestrian attribute recognition: a reliable evaluation under zero-shot pedestrian identity setting. arXiv preprint [arXiv:2107.03576](https://arxiv.org/abs/2107.03576) (2021)
13. Jin, Y., Wang, J., Lin, D.: Multi-level logit distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 24276–24285 (2023)

14. Li, W., Cao, Z., Feng, J., Zhou, J., Lu, J.: Label2label: a language modeling framework for multi-attribute learning. In: Computer Vision-ECCV 2022: 17th European Conference, Tel Aviv, Israel, 23–27 October 2022, Proceedings, Part XII, pp. 562–579. Springer, Heidelberg (2022). https://doi.org/10.1007/978-3-031-19775-8_33
15. Li, Z., et al.: Curriculum temperature for knowledge distillation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, pp. 1504–1512 (2023)
16. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
17. Liu, S., Zhang, L., Yang, X., Su, H., Zhu, J.: Query2label: a simple transformer way to multi-label classification. arXiv preprint [arXiv:2107.10834](https://arxiv.org/abs/2107.10834) (2021)
18. Liu, X., et al.: Hydraplus-net: attentive deep features for pedestrian analysis. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 350–359 (2017)
19. Liu, Y., Shu, C., Wang, J., Shen, C.: Structured knowledge distillation for dense prediction. *IEEE Trans. Pattern Anal. Mach. Intell.* (2020)
20. Liu, Y., Sheng, L., Shao, J., Yan, J., Xiang, S., Pan, C.: Multi-label image classification via knowledge distillation from weakly-supervised detection. In: Proceedings of the 26th ACM International Conference on Multimedia, pp. 700–708 (2018)
21. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: hints for thin deep nets. arXiv preprint [arXiv:1412.6550](https://arxiv.org/abs/1412.6550) (2014)
22. Tan, Z., Yang, Y., Wan, J., Guo, G., Li, S.Z.: Relation-aware pedestrian attribute recognition with graph convolutional networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 12055–12062 (2020)
23. Wang, N., Cui, Z., Li, A., Su, Y., Lan, Y.: Multi-priors guided dehazing network based on knowledge distillation. In: Pattern Recognition and Computer Vision: 5th Chinese Conference, PRCV 2022, Shenzhen, China, 4–7 November 2022, 2022, Proceedings, Part IV, pp. 15–26. Springer, Heidelberg (2022). https://doi.org/10.1007/978-3-031-18916-6_2
24. Wang, Y., Zhou, W., Jiang, T., Bai, X., Xu, Y.: Intra-class feature variation distillation for semantic segmentation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12352, pp. 346–362. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58571-6_21
25. Yang, Y., Qiu, J., Song, M., Tao, D., Wang, X.: Distilling knowledge from graph convolutional networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7074–7083 (2020)
26. Zhang, Y., Xiang, T., Hospedales, T.M., Lu, H.: Deep mutual learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4320–4328 (2018)
27. Zhang, Y., Qin, Y., Liu, H., Zhang, Y., Li, Y., Gu, X.: Knowledge distillation from single to multi labels: an empirical study. arXiv preprint [arXiv:2303.08360](https://arxiv.org/abs/2303.08360) (2023)
28. Zhao, B., Cui, Q., Song, R., Qiu, Y., Liang, J.: Decoupled knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11953–11962 (2022)
29. Zheng, Z., et al.: Localization distillation for dense object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9407–9416 (2022)