Leveraging distillation token and weaker teacher model to improve DeiT transfer learning capability

Christopher Gavra Reswara, Gede Putra Kusuma
Department of Computer Science, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia

Article Info

Article history:

Received Mar 6, 2025
Revised Oct 28, 2025
Accepted Nov 5, 2025

Keywords:

DeiT model
Distillation token
Knowledge distillation
Transfer learning
Transformers architecture
Weak-to-strong generalization

ABSTRACT

Recently, distilling knowledge from convolutional neural networks (CNN) has positively impacted the data-efficient image transformer (DeiT) model. Due to the distillation token, this method is capable of boosting DeiT performance and helping DeiT to learn faster. Unfortunately, a distillation procedure with that token has not yet been implemented in the DeiT for transfer learning to the downstream dataset. This study proposes implementing a distillation procedure based on a distillation token for transfer learning. It boosts DeiT performance on downstream datasets. For example, our proposed method improves the DeiT B 16 model performance by 1.75% on the OxfordIIIT-Pets dataset. Furthermore, we present using a weaker model as a teacher of the DeiT. It could reduce the transfer learning process of the teacher model without reducing the DeiT performance too much. For example, DeiT B 16 model performance decreased by only 0.42% on Oxford 102 Flowers with EfficientNet V2S compared to RegNet Y 16GF. In contrast, in several cases, the DeiT B 16 model performance could improve with a weaker teacher model. For example, DeiT B 16 model performance improved by 1.06% on the OxfordIIIT-Pets dataset with EfficientNet V2S compared to RegNet Y 16GF as a teacher model.

Corresponding Author:

Christopher Gavra Reswara
Department of Computer Science, BINUS Graduate Program - Master of Computer Science
Bina Nusantara University
Jakarta, Indonesia
Email: christopher.reswara@binus.ac.id

1. INTRODUCTION

Recently, transformer [1] architectures have become the model of choice in natural language processing (NLP) and computer vision. Due to the self-attention module, the powerful network capacity with that architecture could perform various tasks. In NLP, transformer models achieve competitive results in developing the large language model (LLM), such as GPT-4o [2], Llama 3.2 [3], and Gemini 1.5 [4]. These LLMs help complete human tasks like text summarization [5], sentiment analysis [6], question answering [7], and others. In addition, the transformer model achieves excellent performance in computer vision, including image classification [8], object detection [9], image matching [10], and other tasks.

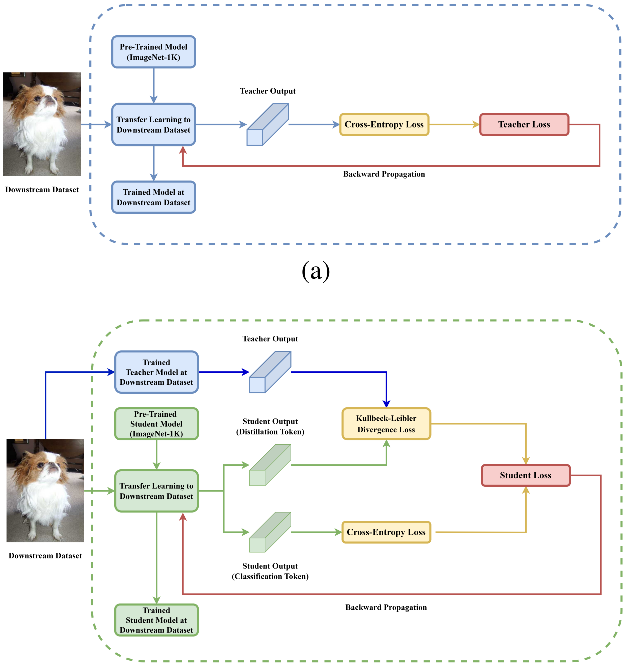
The first transformer-based model in computer vision is the vision transformer (ViT) [11]. That model leverages raw image patches as input and classification tokens as output. Subsequently, transformer-based models in computer vision developed into various models, such as DeiT [12], Swin [8], Swin V2 [13], and others. For example, DeiT introduces a new procedure for knowledge distillation (KD) [14] in a transformer-based model. This model has a new token compared to the ViT, a distillation token. The token is used to calculate the loss between teacher and student output.

To improve DeiT performance, the RegNet Y 16GF [15] model was used as its teacher. RegNet Y 16GF has 83.6M parameters, similar to DeiT B 16, which has 86.6M. While training the student model, the output of the teacher model becomes a supervisor. Furthermore, the classification token of DeiT is computed loss with cross-entropy loss. On the other hand, the distillation token of DeiT and teacher output are computed loss with Kullback-Leibler divergence (KL Divergence) [16]. Both result losses will be average and become student losses used for backward propagation in the student model. Unfortunately, this technique is not yet used for transfer learning to downstream datasets.

Therefore, this study investigated the effects of utilizing a distillation token for transfer learning to a downstream dataset. While the DeiT paper has explored the impact of the distillation token for training a model from scratch, it has not explicitly addressed its influence on utilizing it for transfer learning to a downstream dataset. It is aiming to enhance the transfer learning capability of the transformer-based model. To prove it, we design a simple setup.

In Figure 1, we utilize RegNet Y 16GF to become a teacher model. (a) We transfer learning pre-trained RegNet Y 16GF on ImageNet-1k [17] to the downstream dataset, and the result we called the trained teacher model at a downstream dataset (as shown in Figure 1 (a)). After that, (b) we leverage the trained teacher model at a downstream dataset to supervise pre-trained DeiT on ImageNet-1k as a student model to transfer learning to the same downstream dataset (as shown in Figure 1 (b)). In this way, we prove that this technique could improve DeiT's transfer learning capability. The experimental results of this study use CIFAR-10 [18], CIFAR-100 [18], Oxford 102 Flowers [19], and Oxford-IIIT Pets [20] as downstream datasets.

In addition, we adopt the weak-to-strong generalization [21] concept to simplify (a) transfer learning pre-trained teacher model on ImageNet-1k to the downstream dataset. This concept was based on artificial intelligence's rapid and robust development, especially the transformers-based model. The superalignment model, which is more intelligent than humans, is possible. In contrast, humans need help understanding to ensure the superalignment model is still correct and safe. Therefore, this concept proves that a weaker supervisor model still supervises a robust model.


Pre-Trained Model (ImageNet-1K)
Teacher Output
Transfer Learning to Downstream Dataset
Cross-Entropy Loss
Teacher Loss
Backward Propagation
Downstream Dataset
Trained Model at Downstream Dataset
Teacher Output
Trained Teacher Model at Downstream Dataset
Pre-Trained Student Model (ImageNet-1K)
Student Output (Distillation Token)
Kullbeck-Leibler Divergence Loss
Student Loss
Transfer Learning to Downstream Dataset
Cross-Entropy Loss
Student Output (Classification Token)
Downstream Dataset
Backward Propagation
Trained Student Model at Downstream Dataset

(a)

(b)

Figure 1. Illustration of our proposed method, (a) the pre-trained teacher model is used on the ImageNet-1k dataset for transfer learning to the downstream dataset and (b) the pre-trained student model is used on the ImageNet-1k dataset and the trained teacher model on the downstream dataset

The teacher model used in the DeiT paper is RegNet Y 16GF. That model has 83.6M parameters, similar to DeiT B 16 (student model), which has 86.6M. To implement the weak-to-strong generalization concept, we propose to use a weaker teacher model. We utilize EfficientNet B4 (19.3M) [22] and EfficientNet V2S (21.5M) [23]. We use the teacher model, which is approximately 75% weak compared to the student model. Our contributions are listed as follows:

(a) We propose using a distillation procedure based on a distillation token for transfer learning to the downstream dataset. We find this technique capable of improving DeiT model performance on the downstream dataset.

(b) We introduce using a weaker model as a teacher of the DeiT model. Its method could reduce (a) the transfer learning process on the teacher model because it uses approximately 75% of the weak without reducing DeiT (student) model performance.

(c) We find that the CNN model is the best teacher for the transformers model in the transfer learning process to downstream datasets. In addition, we find that using soft distillation outperforms hard distillation.

2. METHOD

2.1. Dataset

In this study, the ImageNet-1k dataset is used as a large dataset to train a model from scratch. That dataset has 1000 classes and consists of 1,281,167 training images, 50,000 validation images and 100,000 test images. Furthermore, a trained model in the ImageNet-1k dataset will be used for transfer learning to the downstream datasets, i.e. CIFAR-10, CIFAR-100, Oxford 102 Flowers and Oxford-IIIT Pets. Table 1 presents detailed information on downstream datasets.

2.2. Train, validation, and test split data

The dataset is split into a train, validation, and test set for conducted model training. A model utilizes a training set for training. That set will be augmented so that a model could be trained well. In contrast, validation and test sets are not augmentation. A validation set is used to calculate the error rates of a model during training and its impact on the backward propagation process. Meanwhile, a test set is used to evaluate the performance model after the training. In this study, we take a 10% train image of CIFAR-10, CIFAR-100, and OxfordIIT-Pet for validation test. However, we use the default train and validation split of Oxford 102 Flowers, each 50% for the train and validation set. Table 2 shows a detailed split of the dataset in this study.

Table 1. Downstream datasets information

Dataset	Size (Train/Test)	Classes
CIFAR-10	50,000/10,000	10
CIFAR-100	50,000/10,000	100
Oxford 102 Flowers	2,040/6,149	102
OxfordIIIT-Pets	3,680/3,669	37

Table 2. Split dataset to train, validation, and test set

Dataset	Size (Train/Val/Test)
CIFAR-10	45,000/5,000/10,000
CIFAR-100	45,000/5,000/10,000
Oxford 102 Flowers	1,020/1,020/6,149
OxfordIIIT-Pets	3,312/368/3,669

2.3. Data preprocessing

The data preprocessing technique is used for all downstream datasets and all parts of that dataset, as well as train, validation, and test sets. Technique data preprocessing used in this study are resized images, standardization, and normalization. First, all images will be resized to 224 x 224 pixels. Downstream datasets will be standardized to convert the image value from 0.0 to 255.0 into 0.0 to 1.0. Then, convert the image array structure from Height, Width, and Channel to Channel, Height, and Width. After standardization, the downstream dataset will be normalized.

Normalization was conducted based on each downstream dataset average and standard deviation of each channel. Therefore, the normalization value for CIFAR-10 is different from other downstream datasets, such as CIFAR-100. For CIFAR-10, the averages of each channel (Red, Green, Blue) are 0.4914, 0.4822, 0.4465, and the standard deviations of each channel (Red, Green, Blue) are 0.247, 0.243, 0.261. Meanwhile, in CIFAR-100, the averages are 0.5071, 0.4865, 0.4409, and the standard deviations are 0.267, 0.256, 0.276. Oxford 102 Flowers averages 0.4330, 0.3819, 0.2964, and the standard deviations of each channel are 0.273, 0.224, 0.253. Finally, OxfordIIIT-Pets averages are 0.4782, 0.4458, 0.3956, and the standard deviations are 0.247, 0.241, 0.249.

2.4. Data augmentation

The data augmentation technique is only implemented in the train set of all downstream datasets. The techniques used are random crop and random horizontal flip images. For CIFAR-10 and CIFAR-100, the original images are 32 x 32 pixels. Therefore, both datasets will randomly crop to 28 x 28 pixels. Meanwhile, Oxford 102 Flowers and OxfordIII-Pets have a variety of image sizes. Both datasets will randomly crop to 196 x 196 pixels. After that, images will be randomly flipped horizontally. After data augmentation, the train set of all downstream datasets will be data preprocessed.

2.5. Distillation loss

Distillation loss is a numerical metric that measures the difference between the student and teacher models' predicted output. Two techniques were used in this study to compute distillation loss, i.e., hard distillation and soft distillation. Hard distillation uses cross-entropy loss to compute distillation loss, while soft distillation uses the KL divergence function.

Hard distillation. Let Z_s be the logits of the DeiT (student) model and Z_t be the logits of the teacher model. Then, we denote $\mathcal{L}_{CE}$ by the cross-entropy loss and ψ by the softmax function. Especially for hard distillation, the teacher models' predicted output must be computed with the argmax function, which becomes the hard decision of the teacher model and we donated it by $y_t = argmax_c Z_t(c)$. Finally, the function to compute hard distillation loss could be defined as follows:

$$\mathcal{L}_{distill}^{hard} = \mathcal{L}_{CE}(\psi(Z_s), y_t)$$

Soft distillation. We denoted KL be the KL Divergence function and τ as the temperature of the soft distillation. The temperature in the soft distillation is used to smooth the probability distribution. In this study, we used $\tau = 2$ for all soft distillation experiments. The function to compute soft distillation loss could be defined as follows:

$$\mathcal{L}_{distill}^{soft} = \tau^2 KL(\psi(\frac{Z_s}{\tau}), \psi(\frac{Z_t}{\tau}))$$

After computing distillation loss, we can compute global loss for the backward propagation process. Global loss is the average between student model loss and distillation loss. Let $\mathcal{L}_{global}$ by global loss and y by true class. Therefore, the function to compute global loss could be defined as follows:

$$\mathcal{L}_{global} = \frac{1}{2}\mathcal{L}_{CE}(\psi(Z_s), y) + \frac{1}{2}\mathcal{L}_{distill}$$

2.6. Training process

The training process in this study uses AdamW [24] optimizer and CosineLRScheduler [25]. Moreover, the training process was conducted in 10 epochs, with a batch size of 32 and a random seed of 42. The checkpoint model technique was also used during training based on the best validation accuracy. Finally, the training process for transfer learning of the DeiT model to the downstream dataset only uses the attention layer. Other layers were frozen.

2.7. Experiment setup

Based on our proposed method, as shown in Figure 1, we present two steps to improve DeiT transfer learning capability. First, we use a pre-trained model on the ImageNet-1k dataset as a teacher model. That teacher model needs to transfer learning to downstream datasets (Figure 1(a)). The results of the first step are the Trained Teacher model and logits of the teacher model, which we denote by Z_t.

In the next step, we will use a pre-trained DeiT model on the ImageNet-1k dataset. Then, in the transfer learning student model to downstream datasets process, we use the Trained Teacher model as a helper through Z_t. Z_t will be compared with a distillation token using the distillation method and produce distillation loss. That loss will be computed with student loss to become a global loss. It is a loss used by the student model in updating the weight of the student model. In that way, the teacher model could help the student model and improve student model performance.

3. RESULTS AND DISCUSSION

3.1. CNN vs Transformer teacher

First, we proposed using a distillation token to transfer learning to the downstream dataset. Therefore, we observed the best teacher architecture model for transfer learning. Figure 2 compares the performance of the DeiT B model while transferring learning to CIFAR-10, CIFAR-100, Oxford 102 Flowers, and OxfordIIIT-Pets dataset between RegNet Y 16GF and Deit B 16 as a teacher model. We found that using the CNN architecture (RegNet Y 16GF) as a teacher model outperforms the transformer architecture (DeiT B 16).

Inductive bias from CNN adapted to Transformers through distillation makes CNN a better teacher, as explained by Abnar [26]. CNN has a local inductive bias that could help DeiT learn faster, and complementary transformers architecture designed global inductive bias. Hence, RegNet Y 16GF could outperform the transfer learning process only in 10 epochs. This study's following experiment uses a CNN architecture model, specifically RegNet Y 16GF, with 83.6M parameters as a teacher model.

3.2. Hard vs Soft distillation

In addition, we compare two techniques to compute distillation loss, as shown in Figure 3. Soft distillation outperforms in all downstream datasets. For example, transfer learning DeiT B 16 model to Oxford 102 Flowers with soft distillation accuracy is 95.83% compared to hard distillation, only 92.51%. Likewise, the performance of soft distillation is 1.34% higher than that of hard distillation in the CIFAR-100 dataset.

Soft distillation gives information on the predicted probability class of data from the teacher model to the distillation token through KL Divergence. Its token distillation could adjust performance better because the actual class is not always in the first teacher's prediction. Possible actual class data in the second or third of the teacher's predicted. Hence, the following experiment in this study uses the soft distillation technique with $\tau = 2$.

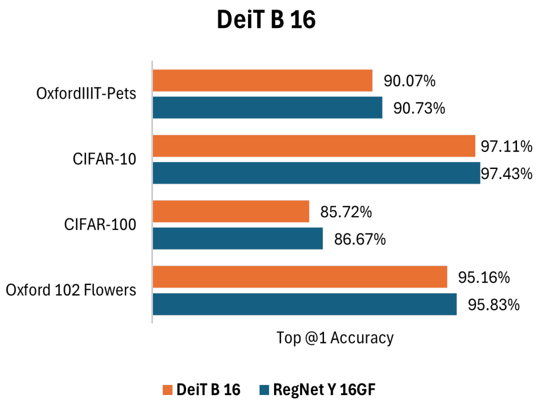


Figure 2. Comparison performance DeiT B 16 model with different teacher architecture. CNN architecture (RegNet Y 16GF) outperforms transformer architecture (DeiT B 16)

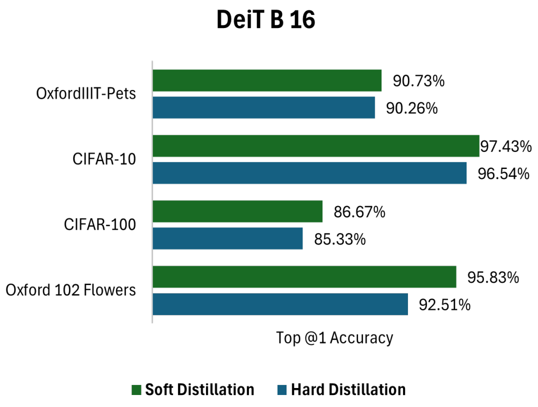


Figure 3. Comparison performance DeiT B 16 model with different distillation loss techniques. Soft distillation outperforms hard distillation

3.3. Transfer learning to downstream datasets

Finally, we have configured RegNet Y 16GF, a teacher model and soft distillation, which makes the DeiT B 16 model better performance. Furthermore, we proved that using distillation tokens and teacher-predicted output to compute distillation loss (our proposed method) is better than just using average distillation tokens and classification tokens (without teacher or standard transfer learning) for transfer learning to downstream datasets. Figure 4 shows that our proposed method significantly improves the DeiT B 16 model. For example, the performance of the DeiT B 16 model increased by 1.75% on the OxfordIIIT-Pets dataset. Similarly, on the Oxford 102 Flowers, DeiT B 16 performance with our proposed method is 95.83% compared to without a teacher, only 94.99%.

This happened because the student model could leverage the teacher's knowledge well. Our proposed method could improve the student model because the distillation token gets information from the teacher model. Thats information makes student models learn more straightforward and faster. The proof is that the standard transfer learning process needs 300 epochs in the DeiT paper, while in this study, only 10 epochs.

3.4. Using weaker teacher model

Unfortunately, our proposed method creates an additional process: standard transfer learning teacher model to a downstream dataset. It makes the transfer learning process of DeiT to the downstream dataset longer. However, we have a fortune because of using CNN architecture as a teacher model. The transfer learning CNN teacher model is faster than the transfer learning Transformers teacher model. Even though the transfer learning CNN teacher model is faster, we still tried to reduce the time standard transfer learning teacher model.

Furthermore, we proposed using a weaker CNN model as a teacher. The model is arguably weaker based on the size of the model parameters. The previous experiment used RegNet Y 16GF as a teacher model with 83.6M parameters. Its model's parameters are similar to the student model, in which DeiT B 16 has 86.6M parameters. Then, we present two weaker models as teachers, i.e., EfficientNet B4 and EfficientNet V2, each with 19.3M and 21.5M parameters, respectively. Therefore, our experiment uses a teacher model that is approximately 75% weaker than a student model, DeiT B 16. Table 3 shows a detailed description of the size of the model parameters.

The result is that EfficientNet V2, whose model size is only 24.82% compared to the DeiT B 16 model, can outperform in CIFAR-10, CIFAR-100, and OxfordIIIT-Pets datasets, as shown in Figure 5. In addition, the performance of the student model with the weaker model is similar to RegNet Y 16GF. For example, in the OxfordIIIT-Pets dataset, the performance of the student model with EfficientNet B4 as a teacher model, whose model size is only 22.82% compared to the student model, decreased by only 0.57% (90.16% vs 90.73%) compared to RegNet Y 16G. Our study shows that a weaker teacher model could simplify the training model for the teacher. Additionally, a weaker model may yield better student model performance in some experiments.

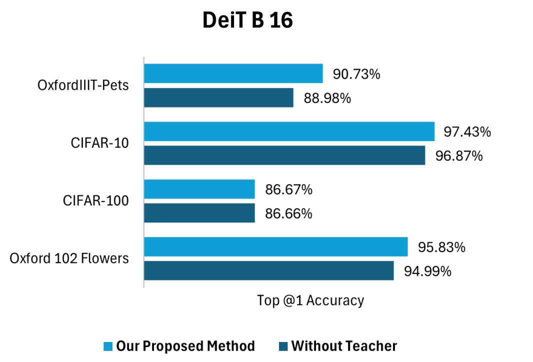


Figure 4. Comparison performance DeiT B 16 model in transfer learning to downstream datasets between our proposed method and the student model without a teacher model (standard transfer learning)

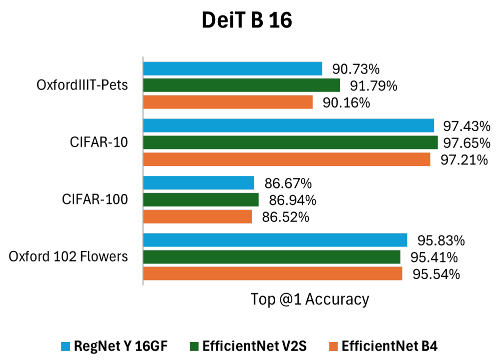


Figure 5. Comparison performance DeiT B 16 model in transfer learning to downstream datasets between RegNet Y 16GF, EfficientNet B4, and EfficientNet V2S

3.5. Using another student model

Finally, we proved that using a distillation token for transfer learning to downstream datasets could improve the DeiT B 16 model performance. Moreover, we also proved that using a weaker model as a teacher model could reduce the complexity of the training teacher model and improve student model performance in several downstream datasets. Therefore, we try the same concept in DeiT S 16 as a student model and EfficientNet B0 as a weaker teacher model to prove that our proposed method applies to the variety model. A teacher model is arguably weaker by the size of the model parameters between the teacher and student model. We denoted DeiT S 16 as the baseline of the model's size (100%). Then, we determine EfficientNet B0 as a teacher model with 5.3 parameters or 24.09% compared to the student model and RegNet Y 16G as a teacher model with 83.6M parameters or 380% compared to the student model. Table 4 shows detailed information on the model size in this experiment.

Like the Deit B 16 model, DeiT S 16 with EfficientNet B0 as a teacher model outperforms compared to RegNet Y 16GF as a teacher model in CIFAR-100, Oxford 102 Flowers, and OxfordIIIT-Pets. Moreover, the EfficientNet B0 model could increase 0.99% the DeiT S 16 performance in the CIFAR-100 dataset, as shown in Figure 6. Conversely, the difference in performance in the CIFAR-10 dataset is only 0.07% (96.89% vs 96.82%) between RegNet Y 16GF and EfficientNet B0 as a teacher. Thus, this experiment proves that our proposed method could improve performance on various models.

Table 3. Model size in DeiT B 16 experiment

Model	Params	Model size
DeiT B 16	86.6M	100%
RegNet Y 16GF	83.6M	96.53%
EfficientNet B4	19.3M	22.28%
EfficientNet V2S	21.5M	24.82%

Table 4. Model size in DeiT S 16 experiment

Model	Params	Model size
RegNet Y 16GF	83.6M	380%
DeiT S 16	22M	100%
EfficientNet B0	5.3M	24.09%

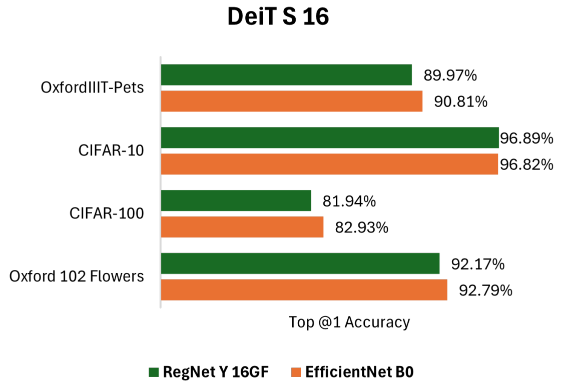


Figure 6. Comparison performance DeiT S 16 model in transfer learning to downstream datasets between RegNet Y 16GF and EfficientNet B0

4. CONCLUSION

Recent observations suggest that a new procedure of KD in the ViT model with distillation tokens can improve performance while training it from scratch. Our findings provide conclusive evidence that this new KD procedure can also enhance model performance when applied to a downstream dataset through transfer learning. Utilizing a distillation token to calculate distillation loss between student output and teacher output remains a helpful technique for the DeiT model in the transfer learning process. The DeiT model (student model) can effectively learn from teacher knowledge. This could happen by supporting CNN architecture as a teacher model and computing the distillation loss using soft distillation.

In addition, we proposed using a weaker teacher model. We present that several downstream datasets could improve the performance of the DeiT model. Otherwise, the performance of the DeiT model with a weaker teacher model is similar to RegNet Y 16GF as a teacher model. However, the complexity of the training teacher model could be decreased by approximately 75%. Therefore, our proposed method of using a weaker teacher model could improve the efficiency of the training process.

Our study demonstrates that utilizing a distillation token and a weaker teacher model can enhance the transfer learning capability of the DeiT model. Thus, future studies may explore the implementation of quantization and pruning methods, allowing the size of the DeiT model parameters to be similar to that of the weaker teacher model. Additionally, it could also be explored to incorporate a distillation token technique into other transformer models, such as Swin and PVT.

FUNDING INFORMATION

Authors state there is no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Christopher Gavra Reswara	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Gede Putra Kusuma	✓	✓		✓						✓		✓		

C	: **C**onceptualization	I	: **I**nvestigation	Vi	: **Vi**sualization
M	: **M**ethodology	R	: **R**esources	Su	: **Su**pervision
So	: **So**ftware	D	: **D**ata Curation	P	: **P**roject Administration
Va	: **Va**lidation	O	: Writing - **O**riginal Draft	Fu	: **Fu**nding Acquisition
Fo	: **Fo**rmal Analysis	E	: Writing - Review & **E**diting		

CONFLICT OF INTEREST STATEMENT

Authors state there is no conflict of interest.

DATA AVAILABILITY

– The datasets analyzed during the current study are publicly available.

– The CIFAR-10 and CIFAR-100 datasets are available at https://www.cs.toronto.edu/ kriz/cifar.html.

– The Oxford 102 Flowers dataset can be found at https://doi.org/10.1109/ICVGIP.2008.47.

– The Oxford-IIIT Pet dataset is available at https://doi.org/10.1109/CVPR.2012.6248092.

REFERENCES

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017, doi: 10. 48550/arXiv.1706.03762.

[2] Y. Wu, X. Hu, Z. Fu, S. Zhou, and J. Li, "GPT-4o: visual perception performance of multimodal large language models in piglet activity understanding," *Arxive*, 2024, [Online]. Available: http://arxiv.org/abs/2406.09781.

[3] Meta, "The Llama 3 Herd of Models," *arXiv*, 2024.

[4] Gemini, "Gemini 1.5: unlocking multimodal understanding across millions of tokens of context," 2024.

[5] H. Shakil, Z. Ortiz, G. C. Forbes, and J. Kalita, "Utilizing GPT to enhance text summarization: a strategy to minimize hallucinations," *Procedia Computer Science*, vol. 244, pp. 238–247, 2024.

[6] J. Šmíd, P. Priban, and P. Kral, "LLaMA-based models for aspect-based sentiment analysis," in *Proceedings of the 14th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, Aug. 2024, pp. 63–70, doi: 10.18653/v1/2024.wassa-1.6.

[7] J. Ding, H. Nguyen, and H. Chen, "Evaluation of question-answering based text summarization using LLM invited paper," in *Proceedings - 6th IEEE International Conference on Artificial Intelligence Testing, AITest 2024*, 2024, pp. 142–149, doi: 10.1109/AITest62860.2024.00025.

[8] Z. Liu *et al.*, "Swin transformer: hierarchical vision transformer using shifted Windows," in *Proceedings of the IEEE International Conference on Computer Vision*, Oct. 2021, pp. 9992–10002, doi: 10.1109/ICCV48922.2021.00986.

[9] Y. Li, H. Mao, R. Girshick, and K. He, "Exploring plain vision transformer backbones for object detection," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13669 LNCS. pp. 280–296, 2022, doi: 10.1007/978-3-031-20077-9_17.

[10] C. Cao and Y. Fu, "Improving transformer-based image matching by cascaded capturing spatially informative keypoints," in *Proceedings of the IEEE International Conference on Computer Vision*, Oct. 2023, pp. 12095–12105, doi: 10.1109/ICCV51070.2023.01114.

[11] A. Dosovitskiy *et al.*, "An image is worth 16X16 words: transformers for image recognition at scale," *ICLR 2021 - 9th International Conference on Learning Representations*, 2021.

[12] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *Proceedings of Machine Learning Research*, 2021, vol. 139, pp. 10347–10357.

[13] Z. Liu *et al.*, "Swin transformer V2: scaling up capacity and resolution," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2022, vol. 2022-June, pp. 11999–12009, doi: 10.1109/CVPR52688.2022.01170.

[14] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network." 2015, [Online]. Available: http://arxiv.org/abs/1503.02531.

[15] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, "Designing network design spaces," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10425–10433, doi: 10.1109/CVPR42600.2020.01044.

[16] S. Kullback and R. A. Leibler, “On Information and Sufficiency,” *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951, doi: 10.1214/aoms/1177729694.

[17] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, “ImageNet: a large-scale hierarchical image database,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009*, 2009, pp. 248–255, doi: 10.1109/CVPR.2009.5206848.

[18] A. Krizhevsky, “Learning multiple layers of features from tiny images.” pp. 32–33, 2009.

[19] M. E. Nilsback and A. Zisserman, “Automated flower classification over a large number of classes,” in *Proceedings - 6th Indian Conference on Computer Vision, Graphics and Image Processing, ICVGIP 2008*, 2008, pp. 722–729, doi: 10.1109/ICVGIP.2008.47.

[20] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. V. Jawahar, “Cats and dogs,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3498–3505, doi: 10.1109/CVPR.2012.6248092.

[21] C. Burns *et al.*, “Weak-to-strong generalization: eliciting strong capabilities with weak supervision,” *Proceedings of Machine Learning Research*, vol. 235. pp. 4971–5012, 2024.

[22] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” *36th International Conference on Machine Learning, ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019.

[23] M. Tan and Q. V. Le, “EfficientNetV2: smaller models and faster training,” *Proceedings of Machine Learning Research*, vol. 139, pp. 10096–10106, 2021.

[24] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” *arXiv preprint arXiv:1711.05101*, 2019.

[25] I. Loshchilov and F. Hutter, “SGDR: stochastic gradient descent with warm restarts,” *arXiv preprint arXiv:1608.03983*, 2017.

[26] S. Abnar, M. Dehghani, and W. Zuidema, “Transferring inductive biases through knowledge distillation,” *Arxiv*, 2020, [Online]. Available: http://arxiv.org/abs/2006.00555.

BIOGRAPHIES OF AUTHORS

Christopher Gavra Reswara received his bachelor’s degree in computer science from Bina Nusantara University, where he is pursuing a master’s degree in the same field. He also works as a Programmer at the Bina Nusantara IT Division. His research focuses on AI, recommendation systems, and computer vision, and he has authored two conference papers on recommendation systems. He can be contacted at: christopher.reswara@binus.ac.id.

Gede Putra Kusuma received Ph.D. degree in Electrical and Electronic Engineering from Nanyang Technological University (NTU), Singapore, in 2013. He is currently working as a Lecturer and Head of Department of Master of Computer Science, Bina Nusantara University, Indonesia. Before joining Bina Nusantara University, he was working as a Research Scientist in I2R – A*STAR, Singapore. His research interests include computer vision, deep learning, face recognition, appearance-based object recognition, gamification of learning, and indoor positioning system. He can be contacted at: inegara@binus.edu.