Pattern Recognition Letters
journal homepage: www.elsevier.com

Improving Knowledge Distillation using Unified Ensembles of Specialized Teachers

Adamantios Zaras[a,**], Nikolaos Passalis[a], Anastasios Tefas[a]

[a]*Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki 541 24, Greece*

ABSTRACT

The increasing complexity of deep learning models led to the development of Knowledge Distillation (KD) approaches that enable us to transfer the knowledge between a very large network, called teacher and a smaller and faster one, called student. However, as recent evidence suggests, using powerful teachers often negatively impacts the effectiveness of the distillation process. In this paper, the reasons behind this apparent limitation are studied and an approach that transfers the knowledge to smaller models more efficiently is proposed. To this end, multiple highly specialized teachers are employed, each one for a small set of skills, overcoming the aforementioned limitation, while also achieving high distillation efficiency by diversifying the ensemble. At the same time, the employed ensemble is formulated in a unified structure, making it possible to simultaneously train multiple models. The effectiveness of the proposed method is demonstrated using three different image datasets, leading to improved distillation performance, even when compared with powerful state-of-the-art ensemble-based distillation methods.

1. Introduction

Deep Learning (DL) models have evolved rapidly over the recent years, leading to state-of-the-art performance. However, DL models typically require an immense amount of parameters, which leads to large and slow models. The advent of powerful dedicated accelerators, e.g., Graphics Processing Units (GPUs) (Chetlur et al., 2014) and Tensor Processing Units (TPUs) (Jouppi et al., 2017), allowed the training of such enormous models, as well as effectively deploying them in many applications. However, deploying DL models in mobile and embedded settings, e.g., on mobile phones, robots, etc., still remains especially challenging due to energy and computational power constraints. These limitations fueled the interest of the scientific community in developing a wide range of methods for reducing the size and complexity of DL models and increasing their speed, without reducing their accuracy. These methods range from replacing computationally intensive operations (Cheng et al., 2015), pruning approaches (Srinivas and Babu, 2015), quantizing the parameters of the models to reduce memory requirements and increase inference speed and/or applying hashing methods (Wu et al., 2016; Peng and Chen, 2019; Peng et al., 2019; Durmaz and Bilge, 2019), as well as developing faster and more lightweight architectures optimized for inference (Iandola et al., 2016; Howard et al., 2017; Luo et al., 2020).

Another very promising research direction is Knowledge Distillation (KD) (Hinton et al., 2015; Romero et al., 2015; Duan et al., 2019). KD works by employing a large, well trained model, known as *teacher*, to guide the training process of another lightweight model, known as *student*. In this way it is possible to *distill* the knowledge encoded in the larger model into a smaller and faster one. Also, note that compared to other methods that aim at reducing the computational requirements for a DL model, e.g., quantization or pruning, KD aims at increasing the accuracy of an existing lightweight architecture. This allows KD to be combined with virtually any of the existing methodologies for developing lightweight DL models and further increasing their accuracy. In this way, it provides the flexibility of choosing the exact size and architecture of the final model that we want to deploy. The effectiveness of KD critically relies on the employed teacher model. For example, having a less capable teacher will lead to less knowledge being available to be transferred to the student, potentially limiting its accuracy. At the same time, it has also been shown that when powerful teachers are used, the distillation efficiency can actu-

**Corresponding author: Tel.: +30-694-888-6792;
e-mail: adamanti@csd.auth.gr (Adamantios Zaras)

ally be reduced (Mirzadeh et al., 2019). More powerful teacher models can typically generate more confident classification decisions, leading to reduced diversity, thus explaining their apparent failure to effectively distill their knowledge. Indeed, it has been demonstrated that using less confident teachers can improve distillation efficiency (Panagiotatos et al., 2019).

The question that naturally arises from the previous observation is whether it is possible to develop a powerful teacher, which is, at the same time, capable of effectively transferring its knowledge to a smaller student model, while maintaining its ability to extract meaningful representations. The main contribution of this work is to propose the specializing of multiple teachers, each to a limited range of skills, in order to overcome the aforementioned limitation. Even though each individual teacher is confident in its own small set of skills, thus achieving high accuracy at them, the ensemble's diversity is achieved by training them in different tasks. In this way, more meaningful representations can be extracted. Note that for the purpose of this paper, each skill corresponds to the ability to recognize one category (class) of data. However, this is without loss of generality, since the proposed method can be also applied on other domains, such as reinforcement learning (Teh et al., 2017).

The proposed method can be better understood by considering the following example. Training a powerful teacher to recognize a set of classes will probably lead to it confidently selecting the correct class most of the time. However, it will not be able to recognize similarities between the input object and the rest of the classes, since it has been trained to suppress the rest of the outputs. Instead, consider an ensemble of three teachers, each one trained in a disjoint set of classes. The teacher that is responsible for recognizing the correct class will again be confident in it. The other two, however, despite being less confident, will still classify the input object, according to their corresponding classes. In this way, the rest of the teachers will provide their *opinion* regarding the similarities of the input object to the classes for which they are responsible. This approach effectively provides a way to extract meaningful representations over the classes at hand, while at the same time employing powerful teacher models. Indeed, as it is experimentally demonstrated using three different image datasets, the proposed method leads to improved distillation performance, even when compared with powerful state-of-the-art ensemble-based distillation methods.

The rest of this paper is organized as follows. First, Section 2 provides a brief overview of related distillation methods and highlights the key differences between them and the proposed method. Then, the latter is analytically derived and discussed in Section 3, while the experimental evaluation is provided in Section 4. Finally, conclusions are drawn and future work is discussed in Section 5.

2. Related Work

There is a considerable amount of literature about KD, describing multiple ways in which it can be performed and different fields wherein it could be applied. As already described in the previous Section, the main motivation for applying KD is to more effectively train a lightweight DL model. KD is always performed between two models, where the first one could be either a single model or even an ensemble of models. In the classical approach (Bucila et al., 2006), the method utilizes an ensemble to label unlabeled data that are then used to train a neural network, thus mimicking the function learned by the ensemble and achieving similar accuracy. This process was then extended in (Hinton et al., 2015), by introducing a temperature parameter in the probability estimation process, in order to extract a more meaningful distribution over the classes for the input samples. As in the classical approach, the extracted distributions are used to train the student model. This seminal approach, which is called "*Knowledge Distillation*", inspired many subsequent applications.

Indeed, KD has been used for many other purposes besides model compression. Papernot et al. (2015) have discovered that we can address security issues in DNNs by using the extracted knowledge of a network in order to improve its own tolerance to adversarial samples. Using KD can also significantly increase the speed and effectiveness of a model's pre-training process (Tang et al., 2015), providing a good starting point at the optimization space for the student. Rusu et al. (2015) successfully transferred the policies learned by large Deep Q-learning networks to smaller ones. More recent evidence (Chen et al., 2017; Li et al., 2018) suggest that KD can also be effectively applied for transferring the knowledge of object detection models, used to learn from noisy samples (Li et al., 2017), improve the performance of low-precision networks (Mishra and Marr, 2017), or even boost self-supervised learning, allowing us to use different models for the pretext and the main task (Noroozi et al., 2018). The large number of KD applications highlights the importance of developing more efficient methods for transferring the knowledge from larger and more complex networks to a smaller one, an area on which current approaches seem to be adversely affected by the *capacity gap* between the models (Mirzadeh et al., 2019).

Several efforts have been made to improve the efficiency of KD. Romero et al. (2015) used the representations of intermediate layers of the learning networks as a hint, in order to assist deep and thin students in the distillation process. Later, Zhang et al. (2017) developed a new framework in which the student learns a projection of the knowledge of a teacher's intermediate layer, while being trained at the same time. Zagoruyko and Komodakis (2016); Song et al. (2018) combined KD with the attention methodology. Radosavovic et al. (2018) used distillation, in order to transfer knowledge from *data* and not from *models* in an omni-supervised learning task. In their analysis, Yang et al. (2018) question the need for a more tolerant teacher, instead of the most accurate one. They report that it is more important for a teacher to produce a smooth distribution over its predictions and conclude that high accuracy with spiked distribution of confidence is not that important, since the student can be more easily over-fitted. Lan et al. (2018) proposed an online distillation framework in which the teacher is being trained and at the same time its knowledge is being distilled to the student. Passalis and Tefas (2018) extended the applications of KD to representation learning tasks through a Probabilistic Knowledge Transfer (PKT) framework. Similarity embeddings (Passalis and Tefas, 2019) were also proposed, which can lead to more general, unsupervised KT and can have many applications, such as cross-domain data exploitation. Yuan et al. (2019) suggested that we can remove the role of the teacher from the KD process and develop a self-learning student. This study differs from the aforementioned ones in that it aims to improve the method by focusing on the teacher, instead of focusing on distillation *per se*. It should be noted that most of these approaches can be readily combined with the proposed one to further improve distillation performance.

To the best of our knowledge, this is the first work which employs an efficient unified ensemble of diversified, task-specialized models in order to overcome the apparent ineffectiveness of distillation, when powerful teachers are used. It is worth noting that Hinton et al. (2015) mentioned in their work that it is possible to create specialized teachers by utilizing smaller datasets enriched with more samples from the classes of their specialty, which also requires each teacher to be separately trained. On the other hand, the proposed method employs an efficient unified ensemble approach that allows for the one-step training of the whole ensemble, without the need of individual datasets. Also, Lan et al. (2018) developed a framework which allows the simultaneous training of all the teachers in an ensemble. However, teachers are unspecialized and trained to predict all the classes, reducing the diversity of the models in the ensemble, which limits the efficiency of KD, as also experimentally demonstrated in Section 4.

3. Proposed Method

The proposed Unified Specialized Teachers Ensemble method, abbreviated as USTE, is presented in this Section. The KD process is briefly introduced in the Background Subsection, while the proposed method is analyzed in the following one. It is worth noting that even though the proposed method has been combined with the plain KD, most of the more advanced distillation approaches described in Section 2, can also be used, potentially further increasing its effectiveness.

3.1. Background

KD was introduced as a model compression framework, which eases the training of deep networks by following a student-teacher paradigm, in which the student is trained according to a softened version of the teacher's output Hinton et al. (2015). This suggests that the learned knowledge of a teacher network is hidden in the soft probabilities of its predictions. Therefore, if we were to teach a student model the way a teacher model "thinks", it would be useful to try and impart these similarities among the classes for each sample and not only the final predictions. In order to efficiently transfer the knowledge encoded in the similarity among different classes, Hinton et al. (2015) also introduced a temperature parameter T in the softmax activation. This enables us to tune the fuzziness of class probability estimations, rendering the output probability distribution less spiky.

More specifically, KD works as follows. Let $\{\mathbf{x}_i | i = 1, \ldots, m\}$ be a set of m training samples with Ψ number of classes, while the notation $N(\cdot) \in \mathbb{R}^\Psi$ is used to refer to the teacher network that extracts Ψ logits, one for each class. To simplify the notation, l_{ij} is used to refer to the j-th logit for the i-th training sample. Then, the probability for the j-th class for the corresponding sample is estimated as:

$$p_{ij} = \frac{\exp(l_{ij}/T)}{\sum_{t=1}^{\Psi} \exp(l_{it}/T)}. \quad (1)$$

Higher temperatures will result in a softer probability distribution, while lower temperatures will result in a sharper probability distribution. When tuned properly, temperature allows for revealing the intra-class similarities for each sample.

The student model $f_{\mathbf{W}}(\cdot)$, where $\mathbf{W}$ refers to its trainable parameters, can be trained as follows. The soft student's probabilities q_{ij} are calculated similarly to (1), while the notation $\hat{y}_{ij}$ is used to refer to the regular ($T = 1$) student's output. Then, the distillation loss is defined by combining the regular cross entropy loss with the aforementioned constraint of "mimicking" the teacher's behavior:

$$\mathcal{L}_{KD} = -\lambda \sum_{i=1}^{m} \sum_{j=1}^{\Psi} p_{ij} \log q_{ij} - (\lambda - 1) \sum_{i=1}^{m} \sum_{j=1}^{\Psi} y_{ij} \log \hat{y}_{ij}, \quad (2)$$

where $\mathbf{y}_i$ is the one-hot encoded ground-truth vector for the i-th training sample and $\lambda \in [0, 1]$ is a user-defined parameter that controls the importance of distillation in relation to normal training for the student.

3.2. Unified Specialized Teachers Ensemble

The proposed method works by compiling an ensemble of teacher models, as shown in Fig. 1. Each teacher is trained on a subset of the available classes, allowing it to be highly *specialized*. At the same time, they can still provide predictions for input samples that belong to classes out of their specialization field, *diversifying* the ensemble. Furthermore, instead of training each model separately, a *unified* one-step training procedure is employed, significantly reducing the computational complexity. As a result, this approach allows for the perspective of the most certain model to prevail, while at the same time permitting a multitude of opinions, leading to richer dark knowledge. The dominant teacher is likely to be one of those whose specialization relates to the correct class and therefore enhances its specialization ability even more through the training process. As a result, we believe that the distribution of the unified ensemble will be more spiked for the controversial classes and may require a higher temperature to transfer knowledge optimally, as experimentally demonstrated in Subsection 4.2.

Let $\{N_k\} = \{N_1, N_2, \ldots, N_D\}$ be the set of D specialized teachers. These teachers are trained on the whole training dataset $\mathbf{x}_1, \cdots, \mathbf{x}_m$, where $\mathbf{x}_i$ denotes the i-th training sample. Also, note that ground truth annotations $\mathbf{y}_i$, which are one-hot-encoded vectors, also exist for each training sample $\mathbf{x}_i$, as explained in the previous Subsection. The output of the k-th specialized teacher is denoted by $p_{ij}^{(k)}$, after passing through a softmax function. Applying the softmax function individually for each model is essential to ensure that their output is normalized prior to the final aggregation. Furthermore, note that the output can

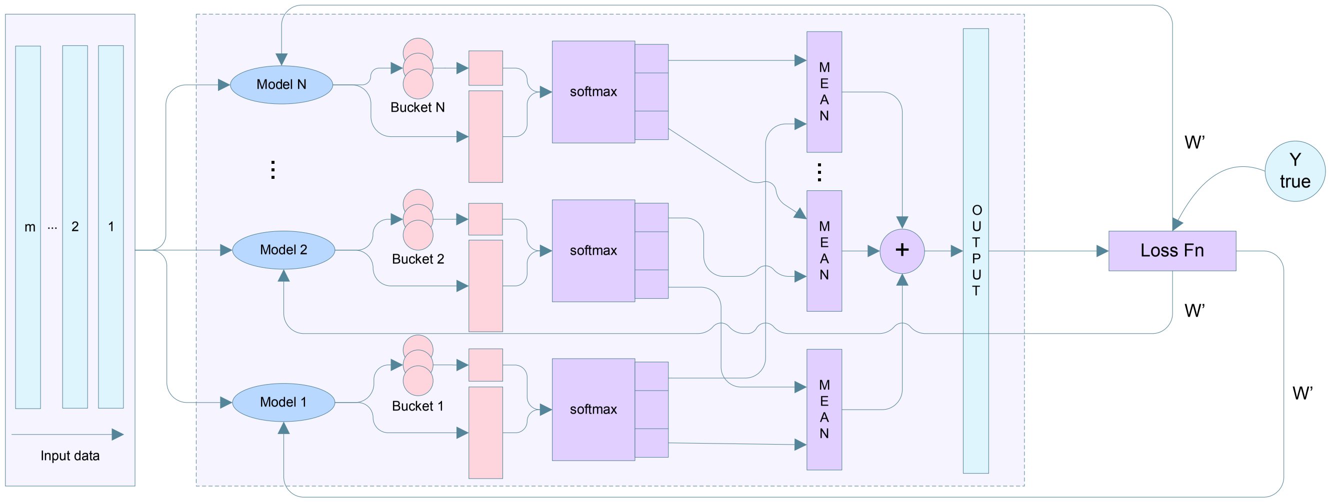

Fig. 1. Unified Specialized Teachers Ensemble structure: The teacher models become separate branches of a large unified network. The large network receives the data as an input and distributes them in every teacher N_k. Subsequently, each teacher N_k predicts the classes of its specialization field, along with an extra bucket class, which represents every other choice, unrelated to its specialization field. The softmax activation function is then applied over each teacher's output in order to produce the normalized probabilities p_i. At this point, the probabilities of the identical classes which have been chosen to be overlapped, are averaged. Finally, the distinct probabilities are aggregated in order to extract the final output distribution of USTE.

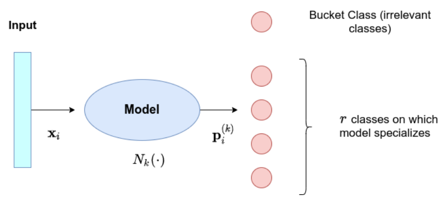

Fig. 2. An individual teacher model. Note that an extra neuron is used, apart from those utilized for the r classes the model predicts. This "bucket" neuron facilitates the effective training of the models with classes that do not belong in their specialization, i.e., the remaining $\Psi - r$ classes.

be softened using the appropriate value for the temperature as described in (1), if needed.

Each specialized teacher predicts a subset of $r = \lceil K\Psi/D \rceil$ classes. The parameter K is called overlapping factor and controls how many times each class will be predicted by a different teacher N_k. The number of times a class is predicted can be calculated as: $K = Dr/\Psi$, assuming that $K\Psi \mod D = 0$. In order to ensure that no two models are specialized in the same classes, they are distributed cyclically over the ensemble. Note that K should be set to an appropriate value so that models do not predict all the available classes, i.e., $K < D$. Furthermore, K should be large enough to ensure that models will not predict one single class, i.e., $K > \lceil D/\Psi \rceil$. Finally, Ψ sets denoted with $\{\Omega(i) | i = 1, \ldots, \Psi\}$ are created, one for each class, and contain $K \in [1, D] \subset \mathbb{N}$ indexes that indicate which teachers participate in the prediction of class i. For example, $\Omega(2) = \{1, 4, 5\}$ symbolizes that the 1-st, the 4-th and the 5-th teachers all predict class 2.

Each teacher is also equipped with an extra "bucket" neuron that is responsible for gathering the predictions of the rest $\Psi - r$ classes, as shown in Fig. 2. This bucket neuron can be used to train each teacher with data that belong to classes out of its expertise. Another advantage of this method is that we can train all the teachers simultaneously by feed-forwarding and back-propagating only one time through the resulting unified architecture. More specifically, the final output of the model is calculated by averaging the K values for each class, as predicted by the individual models. Therefore, the final ensemble's probability estimation for the j-th class and i-th sample is calculated as:

$$p_{ij} = \frac{\exp(a_{ij}/T)}{\sum_{l=1}^{\Psi} \exp(a_{il}/T)}, \tag{3}$$

where

$$a_{ij} = \frac{1}{D} \sum_{t \in \Omega(j)} p_{tj}^{(k)}, \tag{4}$$

and $\Omega(j)$ denotes the set of teachers that predict the j-th class. Note that $p_{tj}^{(k)}$ refers to the neuron of the k-th teacher that predicts the j-th class. As with regular distillation, appropriately tuning the temperature for the ensemble's output is crucial to ensure that the output distribution will not be overly spiked, which can negatively impact the distillation efficiency.

The teacher ensemble model is then directly trained in a unified, one-step fashion to minimize the regular cross-entropy loss:

$$\mathcal{L}_t = \sum_{i=1}^{D} \sum_{j=1}^{\Psi} y_{ij} \log \hat{y}_{ij}, \tag{5}$$

where $\hat{y}_{ij}$ refers to the output of the teacher ensemble with $T = 1$. Note that the whole ensemble can be directly trained, since only one forward and backward pass is required to update the parameters of all the employed models. On the other hand, the student model is trained to minimize the combined distillation loss $\mathcal{L}_s$, as described in (2), where the teacher ensemble model is used to provide the training targets. The Adam algorithm Kingma and Ba (2014), with the default training hyper-parameters, is used for the optimization in this paper. Note that the loss $\mathcal{L}_t$ is minimized by updating the parameters of the teachers, while the loss $\mathcal{L}_s$ is minimized by updating the parameters of the student.

4. Experimental Evaluation

First, the datasets used for evaluating the proposed method are briefly introduced, along with the employed network architectures. Next, the evaluation results are provided and discussed.

4.1. Datasets and Evaluation Setup

The proposed method was evaluated using three different datasets: CIFAR-10, CIFAR-100 Krizhevsky (2012) and Fashion-MNIST Xiao et al. (2017). A tuning phase was performed for setting the hyper-parameters described below, in which the methods depend on, to ensure that the best performance was achieved.

The CIFAR-10 dataset consists of $60,000$ 10-class images, 32×32 in size and is divided into $50,000$ training data and $10,000$ test data. Five teachers that consist of three blocks are used. Each block is composed of two convolutional layers with the same number of filters, which are doubled on each consecutive block (32/64/128 filters). The convolutional layers are followed by a max pooling and among them, batch normalization is used. After every block, a dropout layer is used, with an incremented probability of turning a neuron off each time, which does not exceed 50%. All the convolutional layers are being l_2 regularized. In order to introduce some diversity among the teachers, we use a *ReLU* activation function in two models and *eLU* in the rest of them Clevert et al. (2015), while at the same time we fluctuate the weight decay (ranging from $1e-4$ to $1e-7$) that is used for the l_2 regularization. The student that is used, consists of two blocks and was built following the same methodology.

The CIFAR-100 dataset consists of $60,000$ 100-class images, 32×32 in size and is divided into $50,000$ training data and $10,000$ test data. For the CIFAR-100, the same architectures were used after adding one additional block (with 256 filters). Finally, the Fashion-MNIST dataset consists of $60,000$ 10-class images, 28×28 in size and is divided into $60,000$ training data and $10,000$ test data. For the experiments conducted with the Fashion MNIST dataset, the same architecture with the CIFAR-10 teachers/students was used, but only one convolutional layer was kept per block. All the models were trained for 150 epochs using a learning rate of $1e-4$, which was scheduled to be reduced, multiplying it by 0.4 for each 8 consecutive epochs that showed no improvement in the 3-rd decimal place and a minimum possible value of $5e-6$. A mini-batch of 64 samples was used for all the conducted experiments.

The baseline accuracy among the different trained models is reported in Table 1. Note that apart from the accuracy of the individual models, the ensemble accuracy is also reported. The student was also trained normally, using the same hyperparameters with the teachers, in order to compare the results with that of KD. In order to transfer the knowledge, a temperature $T = 6$ was used and a $\lambda = 0.9$ for CIFAR-10, $T = 2$ and $\lambda = 0.6$ for CIFAR-100, $T = 8$ and $\lambda = 0.6$ for Fashion-MNIST. The knowledge was transferred for 150 epochs, with a learning rate of $1e-3$, which was scheduled to be halved, for each 8 consecutive epochs that showed no improvement in the 3-rd decimal point and a minimum possible value of $1e-8$. A mini-batch of 64 samples was used.

The proposed method was also compared to four other approaches:

1. "Best Teacher": Five individual teacher models were trained and the best of them was used to perform regular KD to the student model.
2. "Ensemble": The knowledge contained in an ensemble of five teachers was directly transferred to the student model using KD, after averaging their output predictions.
3. "Unified Ensemble": The approach proposed in (Lan et al., 2018), was employed to train a unified ensemble with unspecialized teachers and then the knowledge was transferred from this ensemble to the student model.
4. "Specialized Ensemble" ("Special. Ensemble"): Training individual specialized models using the proposed class distribution approach (but without using a unified model structure).

For the proposed method we used $D = 5$ teachers, while the replication factor was set to $K = 2$. To ensure a fair comparison between the evaluated methods, the same student network was used for all the conducted experiments with the same dataset.

4.2. Experimental Results

The evaluation results using the CIFAR-10 dataset are reported in Table 2 from which several conclusions can be drawn. First, note that using plain distillation ("Best Teacher") indeed improves the accuracy of the student, increasing it to 84.28% from 82.19% (baseline student). Using the ensemble of the different teachers further increases the classification accuracy to 84.90%. Quite interestingly, employing a unified ensemble, apart from faster training, allows to also slightly increase the effectiveness of the distillation process. We hypothesize that this happens due to the implicit diversification that emerges through the training process. That is, in the unified ensemble, a few confident models are enough to correctly classify an input sample, allowing for an implicit specialization to emerge among different models. Moreover, when this specialization is induced explicitly, through the specialized ensemble, accuracy further improves. Finally, the best results are acquired when the proposed USTE approach is employed, outperforming plain distillation by about 2% and unified ensemble approach by about 1% (relative increase).

Furthermore, we conducted additional experiments to evaluate the effect of the different ensembling strategies that were employed. The experimental results are reported in Table 3. For these experiments we used 100 images of the CIFAR-10 dataset and averaged the inference time for the different models. An interesting observation is the fact that the proposed USTE method is as fast as the other methods even though it

Table 1. Evaluating the accuracy of different teachers, student and ensembling approaches

Dataset \ Method	Student	Teacher 1	Teacher 2	Teacher 3	Teacher 4	Teacher 5	Ensemble	Unified Ensemble
CIFAR-10	82.19	84.17	84.45	83.72	85.65	85.63	87.10	84.47
CIFAR-100	61.57	59.43	60.44	58.28	60.54	63.14	64.36	59.26
Fashion-MNIST	88.49	92.08	92.33	92.42	92.02	92.10	92.99	92.89

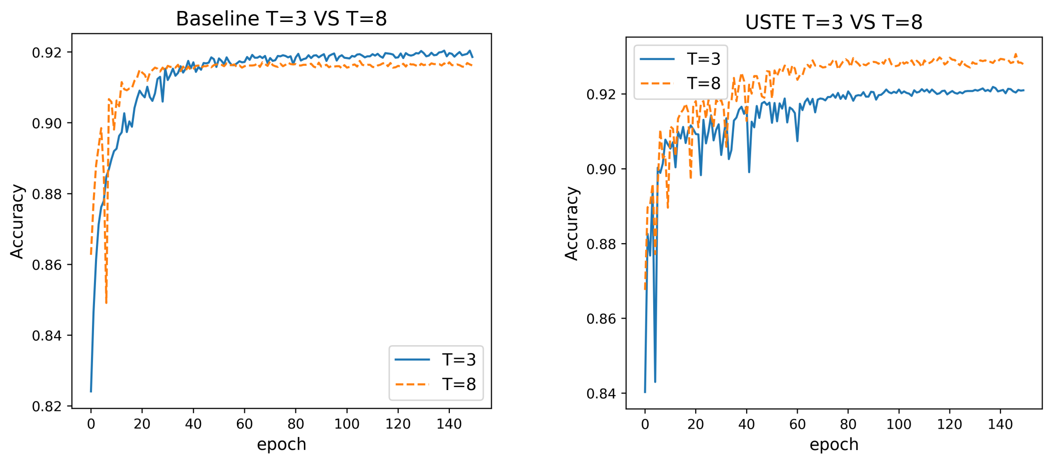

Fig. 3. Effect of raising the temperature with Baseline and USTE in Fashion-MNIST

Table 2. Comparison between different distillation approaches on three different datasets

Method	CIFAR-10	CIFAR-100	Fashion MNIST
Best Teacher	84.28%	64.61%	91.26%
Ensemble	84.90%	65.70%	91.75%
Unified Ensemble	85.03%	66.41%	92.00%
Special. Ensemble	85.43%	66.73%	92.70%
USTE	**85.90%**	**67.14%**	**93.07%**

Table 3. Inference time evaluation between different ensembling methods

Method	**Inference Time**
Ensemble	0.036 s
Unified Ensemble	0.035 s
Specialized Ensemble	0.032 s
Proposed (USTE)	0.032 s

can lead to more accurate models. This phenomenon can be explained, since the number of parameters remains the same and the main difference is the way that the weights are distributed to different submodels. It is also worth noting that the accuracy achieved by the employed architecture is lower than the state-of-the-art models (Huang et al., 2017). However, these more complicated models are difficult to deploy in most mobile and embedded architectures, e.g., NVIDIA Jetson-based processors, especially when multiple DL models must be executed in parallel and there are requirements for real-time and high resolution inference (Tzelepi and Tefas, 2020). In these cases, that often occur in real deployments, the proposed method can provide significant performance benefits compared to the rest of the evaluated distillation strategies.

Similar conclusions can be drawn for the other two datasets (CIFAR-100 and Fashion MNIST). For example, USTE improves the accuracy by 2.8% over plain distillation and about 1% over unified ensemble approach for CIFAR-100 dataset. These results once again confirm that a diversified and specialized teachers' ensemble helps to transfer knowledge better and that unified training leads to better results than training the models individually. It is worth noting that, the results of Table 1, also confirm the hypotheses reported in (Yang et al., 2018), i.e., that classification accuracy is not the major goal of the teacher network when used for KD. Indeed, they report in their work that *".. although this harms the accuracy of the teacher network, it indeed provides more room for the student network(s), and eventually, the students are better than those educated by a strict teacher."*. The proposed method builds upon these observations, providing efficient and diversified teachers that are better suited for the task of KD.

Another question that arises is the effect of the number of teachers used to transfer the knowledge to the performance of the employed method. Therefore, we ran the same experiments using 3 and 7 teachers. The experimental results for three different datasets and numbers of teachers are reported in Table 4. The number of teachers used can have a crucial role in the performance of all the evaluated methods. Increasing the number of teachers indeed increases the effectiveness of knowledge transfer. However, after a certain point, e.g., around 5 teachers, the