International Journal of Computing and Digital Systems
2024, VOL. 17, NO. 1, 1–10

http://dx.doi.org/10.12785/ijcds/XXXXXX

ModReduce: A Multi-Knowledge Distillation Framework

Yahya Abbas[1], Abdelhakim Badawy[1], Mohamed Mahfouz[1], Samah Hussein[1], Samah Ayman[1], Hesham M. Eraqi[2,*] and Cherif Salama[1,3]

[1]*Department of Computer Science & Engineering, The American University in Cairo, Cairo, Egypt*
[2]*Amazon, Seattle, USA*
[3]*Faculty of Engineering, Ain Shams University, Cairo, Egypt*

Abstract: Deep neural networks have achieved revolutionary results in several domains; nevertheless, they require extensive computational resources and memory footprint. Research has been conducted in the field of knowledge distillation, aiming to enhance the performance of smaller models by transferring knowledge from larger networks, which can be categorized into three main types: response-based, feature-based, and relation-based. Existing works explored using one or two knowledge types; however, we hypothesize that distilling all three knowledge types should lead to more comprehensive transfer of information and would improve the student's accuracy. In this paper, we propose ModReduce; a unified knowledge distillation framework that utilizes the three knowledge types using a combination of offline and online knowledge distillation. ModReduce is a generic distillation framework that utilizes state-of-the-art methods for each knowledge distillation type to learn a better student. As such, it can be updated with new state-of-the-art methods as they become available. During training, three student instances each learn a single knowledge type from the teacher using offline distillation before leveraging online distillation to teach each other what they learned; analogous to peer learning in real life where different students can excel in different parts of a subject they are learning from their teacher and then help each other learn the other parts. During inference, only the best performing student is used, so no additional inference costs are introduced. Extensive experimentation on 15 different Teacher-Student architectures demonstrated that ModReduce produces a student that outperforms state-of-the-art methods with an average relative improvement up to 48.29% without additional inference cost. Source code is available at https://github.com/Yahya-Abbas/ModReduce.

Keywords: Knowledge Distillation, Model Compression, Deep Learning, Response Knowledge, Relational Knowledge, Feature Knowledge, Image Classification

1. INTRODUCTION

The term knowledge distillation was formally popularized in [1], and it refers to transferring knowledge from a large pre-trained model to a smaller one, aiming to retain comparable performance to the large model. Knowledge distillation has been receiving increasing attention from the research community due to its promising results in diverse fields such as autoregressive language models [2], visual representation [3], and image-text retrieval [4].

The methods for knowledge distillation vary widely based on several factors like knowledge type, the distillation algorithm, and the teacher-student architecture [5]. Response-based knowledge uses the large model's logits as the teacher model's knowledge. The main idea is that the student optimizes its training over the soft targets, or the softened probability distribution, produced by the teacher model instead of using discrete labels [1]. By mimicking the probabilities the teacher has for incorrect classes, the student model learns better how to generalize [6]. While this method showed great success, one of its major drawbacks is that it disregards the knowledge a teacher model retains in its intermediate layers. This encouraged researchers to introduce methods that capture the knowledge in the intermediate layers of the teacher model, feature knowledge. Feature-based algorithms focus on the features of the teacher model's intermediate layers to guide the student's learning. The challenge is that the teacher and the student models have different abstraction levels, which makes it one of the objectives of the distillation process to determine the best layer associations for maximum performance [7], [8], [9]. Relational-based methods focus on the relationships between different data instances and different activations and neurons [5].

Several algorithms and methods have been introduced, focusing on distilling one or two of the knowledge sources from a teacher model to a student model. While they

E-mail address: yahya-abbas@aucegypt.edu, abdelhakimbadawy@aucegypt.edu, Mohamed.Mahfouz@aucegypt.edu, hysamah@aucegypt.edu, samahayman@aucegypt.edu, heraqi@amazon.com, cherif.salama@aucegypt.edu

** The work was conducted prior to Hesham Eraqi joining Amazon*

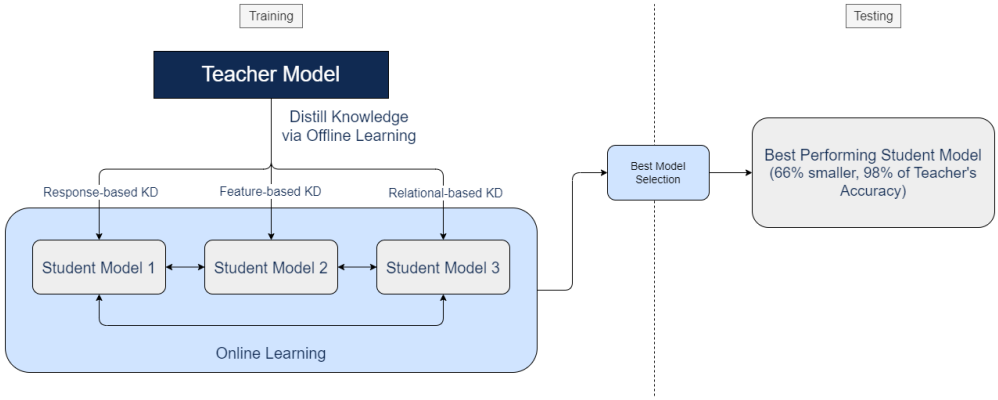


Figure 1. Our ModReduce framework effectively distills the three knowledge types from the teacher model into three student models through offline knowledge distillation. Student models then engage in online distillation to share their learned knowledge and the best-performing student model is elected for inference. On average, the ModReduce-trained student models achieve **98%** of the teacher's accuracy while being **34%** its size, outperforming the underlying state-of-the-art methods.

show promising results, no research addresses the issue of distilling all three knowledge types. To the best of our knowledge, ModReduce is the first work to explore this area. Moreover, we explore combining offline and online distillation strategies to leverage the benefits of both. For online learning distillation, we have explored four different techniques for knowledge aggregation: Peer Collaborative Learning (PCL) [10], On-the-fly Native Ensembling (ONE) [11], Fully Connected Layers (FC), and weighted averaging.

Our main contributions are:

1) We propose ModReduce, a generic plug-and-play multi-knowledge distillation framework that helps transfer the three main types of knowledge from a teacher to a student leveraging the benefits of offline and online distillation and state-of-the-art (SOTA) distillation methods for each knowledge type.
2) ModReduce is a new simple training scheme for knowledge distillation that is extensible and can be updated with new SOTA methods for different knowledge types as well as incorporate novel online distillation techniques in a simple manner.
3) We empirically find that just combining the loss functions of SOTA distillation methods for each knowledge type does not lead to a better student, which further motivated our choice of training scheme.
4) Extensive experimentation on a comprehensive benchmark that is a combination of the benchmarks used in [12], [13] shows ModReduce generally improves the student performance over the underlying methods used.

The rest of the paper is organized as follows; Section 2 explores related work while focusing on the different knowledge types a model possesses and the different knowledge distillation schemes. Section 3 provides an in-depth explanation of our ModReduce framework, covering its structure, the offline and online training phases, the methods implemented, and the overall training algorithm. Section 4 presents our experimental setup along with the key questions we aim to address, while Section 5 covers the results and discussion. Finally, Section 6 concludes the paper and Section 7 offers some thought on future directions.

2. RELATED WORK

Now, we explore the related work that forms the backbone of our framework. This section is organized into two key areas: the types of knowledge within a models and the distillation schemes used to transfer this knowledge.

A. Knowledge Sources

By knowledge we refer to the information that a neural network has grasped during its training, or more abstractly, the learned mapping from inputs to outputs. For instance, a probability distribution resulting from a classifier fed with an image of a rat is closer to a probability distribution of a rabbit than it is to a dog [1]. This means that the network holds this type of knowledge. We examine the different knowledge types that serve as primary targets for distillation as specified in [5], namely, response-based, feature-based, and relation-based knowledge.

Response-Based knowledge is defined as the response of a neural network whenever it is presented with a particular input. The most popular response-based knowledge distillation scheme in image classification tasks is using "soft targets," where the aforementioned probability distribution is softened using a temperature factor T. According to [1], these soft targets provide informative dark knowledge from the teacher model and the probabilities a teacher assigns to wrong classes provide helpful information on how it generalizes. Hinton's soft targets approach is the most popular approach with the best results so far. Unfortunately, response-based knowledge is blind to the inner features in the hidden layers of the model, as it only focuses on the final outputs.

Feature-Based knowledge extends on the idea of response-based knowledge and takes the outputs of intermediate layers into consideration. Accounting for the outputs of the intermediate layers is important because deep neural networks can learn features with different levels of abstraction. For instance, a deep CNN can learn abstract features like straight and curved lines in the shallowest layers while detecting features with higher complexity at the deeper layers [14]. This idea is useful for constructing teacher-student architectures since this type of knowledge can be used in the training of the student network. Many knowledge distillation techniques use a distillation loss function that accounts for feature-based knowledge. Equation 1 represents a general form of the distillation loss function for feature-based knowledge, where $f_t(x)$ and $f_s(x)$ are the feature maps of the teacher and student models respectively; ϕ_t and ϕ_s represent the transformation function of the teacher and student models feature maps and $l_f(.)$ is a similarity measure between the feature maps.

$$L_{FeaD}((f_t(x), f_s(x)) = l_f(\phi_t(f_t(x)), \phi_s(f_s(x))) \quad (1)$$

The state-of-the-art method in feature knowledge distillation was introduced in [12], which aimed to match the semantics between the teacher and student. They introduced semantic calibration for cross-layer knowledge distillation that made better use of the intermediate knowledge by matching the semantic level of the transferred knowledge. Then they used an attention mechanism to automatically learn a soft layer association with multiple targets, which helped the student model in learning from multiple semantically matched hidden layers instead of just one fixed layer.

Relation-Based knowledge captures the interrelations between data samples. Several techniques have been proposed to capture the relations between the training data. Instance Relationship Graph (IRG) [15] introduced a methodology of knowledge distillation based on constructing a graph where features are represented as vertices and relations as edges. Relational Knowledge Distillation (RKD) [16] proposed measuring the relations between training data instances using distance-wise and angle-wise losses that penalize structural differences in relations. Contrastive Representation Distillation (CRD), the current state-of-the-art in relational knowledge distillation [13], captures important structural knowledge of the teacher network. It trains a student to capture significantly more information in the teacher's representation of the data using objective contrastive learning, which encourages the student to map similar inputs to close representations, in some metric space, while mapping different inputs to distant representations.

B. Distillation Schemes

After examining the different knowledge types, we discuss the distillation schemes utilized to transfer this knowledge from teacher to student models, with a focus on offline and online distillation approaches.

Offline Distillation is the most basic and popular kind of distillation. It was introduced alongside the concept of distillation by [1]. This scheme transfers knowledge from a pre-trained expert teacher model into a student model. The whole training process takes place in two phases; first, the training of the teacher model on the set of training samples before distillation. The second phase is extracting knowledge from the teacher model and passing it to the student model. The knowledge is extracted from features, responses, or relations as mentioned in section 2-A. Offline distillation is simple and straightforward to implement as it employs one-way knowledge transfer from a trained teacher. In addition, the student model is usually smaller in size and simpler to train.

On the other hand, in this scheme, the model capacity gap always exists and cannot be avoided due to the difference in complexity between the teacher and student models. *Model capacity* could be defined as a measure of a DNN size based on the number of nodes and layers. The offline distillation methods focus on improving knowledge transfer in several aspects. One aspect is regarding the design of the student model and the knowledge type. For instance, in [17], the student model is deeper than the teacher model but it is much thinner at the same time. Hints from the inner hidden layers of the teacher model are taught to the student model to guide the training process. Another aspect is the loss functions for features or distributions matching.

Online Distillation In the online distillation scheme, both the teacher and student models are being trained and updated simultaneously. Online learning was shown to improve the generalization ability of a network by training it simultaneously with a pool of other networks. Moreover, online learning supports heterogeneity in student networks as they can vary in architecture and size. Several techniques have been proposed for the online learning scheme, such as Peer Collaborative Learning (PCL), On-the-fly Native Ensemble (ONE), and weighted averaging.

Peer Collaborative Learning integrates online ensembling and network collaboration into a unified framework [10]. PCL constructs a multi-branch network for training, in which each branch is called a peer. Multiple random augmentations are performed on the inputs to peers and the feature representations outputted are assembled with an additional classifier as the peer ensemble teacher. Moreover, PCL employs the temporal mean model of each peer as the peer mean teacher to collaboratively transfer knowledge among peers, which helps each peer to learn richer knowledge and facilitates optimizing a more stable model with better generalization.

In ONE, a single multi-branch network is trained while simultaneously establishing a strong teacher on-the-fly to enhance the learning of the target network [11]. The auxiliary branches share the low-level layers with the target network, with each branch, together with the shared layers, acting as an individual model. The ensemble of

Algorithm 1: ModReduce Algorithm

Input: $Teacher$: Pretrained Teacher Model
S: [Response Student, Feature Student, Relation Student]
w_1: Online loss weight
w_2: Offline loss weight
Output: Trained Student Model
Initialize *student* for all $student \in S$
for *epoch* ***in*** *epochs* **do**
  for *batch* ***in*** *training_data* **do**
    $t\_prediction \leftarrow Teacher.predict(batch)$
    for $student \in S$ **do**
      $student.predictions \leftarrow student.predict(batch)$
      $student.offline\_loss \leftarrow student.compute\_offline\_loss()$
    $group\_output \leftarrow compute\_group\_output([student.predictions \mid student \in S])$
    for $student \in S$ **do**
      $student.online\_loss \leftarrow compute\_online\_loss(student.predictions, group\_output)$
      $student.total\_loss \leftarrow w_1 \cdot student.online\_loss + w_2 \cdot student.offline\_loss$

those branches builds the teacher model. The training is performed in a closed loop fashion where the teacher aggregates knowledge from branch models on-the-fly, and this knowledge is distilled back to the branches to enhance the models' learning. Evaluations of ONE report enhancement of the generalization performance while maintaining the computational efficiency.

3. ModReduce

In this section, we explain our contribution—the ModReduce framework. We begin by discussing the overall approach of ModReduce and then we detail the specific implementation steps, including the offline and online training phases as well as the techniques and methods employed.

A. Framework Structure

The structure of our framework, as shown in Figure 1, has one cumbersome teacher model that is pre-trained and has high accuracy for the task at hand, and three identical untrained student models. The framework operation is divided into two phases that are performed consecutively at each training step, the offline distillation phase and the online distillation phase. First, offline distillation is performed between the teacher model and each student model separately, using different knowledge sources for each student. This maximizes the knowledge that each student retrieves from its teacher. Then, online distillation is performed among the student models to aggregate the knowledge gained in the offline phase and enhance generalization. For online learning, we implemented different techniques as mentioned in section 2-B.

B. Offline Training Setup

This section will go through the functionality and loss calculation of each offline distillation method used in our algorithm. We used Hinton for response distillation, SemCKD for feature distillation, and CRD for relational distillation.

Hinton By "Hinton" here we refer to the vanilla KD, which depends on the loss calculated between the logit layers of the teacher and student models. This method uses the outputs of the final softmax layer of the teacher, which contains the probabilities for each class (in a classification task), then applies a temperature for these probabilities to convert them into the soft targets. Equation 2 shows how to obtain a soft target q_i, where T is the softening temperature, i is the index of the class, z_i is the logit computed for class i, and q_i is the probability of class i. If T is greater than 1, we obtain q_i values that are softened probabilities (i.e., soft targets).

$$q_i = \frac{e^{\frac{z_i}{T}}}{\sum_j e^{\frac{z_j}{T}}} \tag{2}$$

SemCKD As a feature knowledge distillation method, SemCKD is concerned with transferring knowledge from the intermediate layers of the teacher to the students. Moreover, it employs an attention mechanism to solve the problem of semantic mismatch caused by the difference in teacher and student architectures which could lead to a degradation in performance. The attention mechanism automatically assigns layers from the teacher model for student layers to learn from. In addition to the attention mechanism, each student layer learns from multiple layers in the teacher model to add cross-layer supervision [12].

CRD As for relational, also known as structural knowledge, CRD is the current state-of-the-art. The original response-based knowledge transfer proposed by [1] ignores the complex interdependencies between the data instances, a problem CRD tries to solve by leveraging a contrastive loss to capture the output correlation [13].

Teacher	Student	Accuracy (%)			
		Hinton	SemCKD	CRD	ModReduce
WRN-40-2 (75.61)	WRN-16-2 (73.26)	75.39	75.10	**76.12**	75.44
	WRN-40-1 (71.98)	74.21	73.11	**74.91**	74.84
ResNet110 (74.31)	ResNet20 (69.06)	70.99	70.95	71.35	**72.01**
	ResNet32 (71.14)	73.66	73.47	73.65	**74.34**
ResNet56 (72.34)	ResNet20 (69.06)	71.70	70.91	71.71	**71.99**
ResNet32x4 (79.42)	ResNet8x4 (73.09)	74.32	75.55	74.97	**75.78**
VGG13 (74.64)	VGG8 (70.46)	73.62	74.08	74.39	**74.64**
Avg.		73.41	73.31	73.87	**74.15**

TABLE I. Top-1 test accuracy (%) on CIFAR-100 of Hinton, SemCKD, CRD, and our ModReduce for similar teacher and student architectures. The results indicate that a ModReduce-trained student generally outperforms its underlying SOTA methods, demonstrating our framework's efficiency in enhancing the student model's performance

C. Online Training Setup

The online learning phase enhances the generalization of the student models through sharing the knowledge gained by the other models in the cohort during the offline phase. We have explored four different online learning techniques inspired from different sources to find the best one for our goal. These four techniques are PCL, ONE, FC, and weighted averaging.

PCL This technique was inspired by [10]. In it, the students try to learn collaboratively from each other by employing a temporal mean model copy as a representation for each student. In the online learning phase, each student tries to mimic the soft logits of the temporal mean models of its peers.

ONE In this technique, inspired by [11], inputs and predictions are used to learn a weight for each student. Those weights are used to produce a group output from the individual student predictions, which is then used as a guide for the students to mimic.

FC Similar to ONE, this technique tries to produce a better group output from individual student predictions for the students to follow. However, this objective is achieved here by employing a dense layer to aggregate the three individual predictions into a single one.

Weighted Averaging This technique has the same objective as ONE and FC, sharing knowledge between the students by aggregating their predictions into a group output that each student is penalized against. As its name suggests, we here try to learn a weight for each student, with the weights being from 0 to 1 and having a total sum of 1. Such goal is achieved by having a learnable weight for each student, and those weights are optimized based on students' performance.

Algorithm 1 details how our ModReduce framework is used to train students.

4. Experimental Setup

To verify our proposed hypothesis and demonstrate the effectiveness of our novel framework, we designed a comprehensive experimental setup to answer a set of questions whose answers shaped how ModReduce works.

Does combining the three losses into one have better results? We began by testing this hypothesis to make sure we do not introduce additional complexity in training if it is not needed. As we discuss and elaborate further in the results section, our initial experiments indicated that simply combining the losses of the different knowledge types in one loss does not produce better results than distilling a single type. This necessitated further experiments to explore different ways of aggregating the different knowledge sources.

Does online learning offer any improvement? Then, we examined whether changing the way we aggregate knowledge sources could enhance performance. We adopted online learning as a way for a cohort of different student to aggregate and share the knowledge they learned with each other. The results for this set of experiments confirmed our revised hypothesis and produced better results than just using one or two knowledge types for training the student.

Which online learning technique do we use? To further investigate which online learning technique achieves the best performance, we integrated four different online learning algorithms into ModReduce: Peer Collaborative Learning (PCL), Fully Connected layers (FC), On-the-fly Native Ensemble (ONE), and Weighted Averaging. This allowed us to identify which online learning strategy is the most effective for our framework.

Comparative Analysis with underlying SOTA methods After proving online learning is better than simply

Teacher	Student	Accuracy (%)			
		Hinton	SemCKD	CRD	ModReduce
ResNet32x4 (79.42)	ShuffleNetV1 (70.50)	74.59	**77.21**	75.77	76.96
	ShuffleNetV2 (72.60)	75.73	78.07	76.57	**78.23**
	VGG8 (70.46)	72.48	75.02	73.68	**75.21**
	VGG13 (74.64)	77.21	79.14	77.71	**79.51**
VGG13 (74.64)	ShuffleNetV2 (72.60)	75.89	76.24	76.26	**76.76**
	MobileNetV2 (64.60)	68.72	68.66	**69.66**	69.23
WRN-40-2 (75.61)	MobileNetV2 (64.60)	69.02	69.77	**70.13**	69.37
	ShuffleNetV1 (70.50)	75.45	76.93	76.59	**77.14**
Avg.		73.64	75.13	74.55	**75.30**

TABLE II. Top-1 test accuracy (%) on CIFAR-100 of Hinton, SemCKD, CRD, and our ModReduce for different teacher and student architectures. The results indicate that a ModReduce-trained student generally outperforms its underlying SOTA methods, demonstrating our framework's efficiency in enhancing the student model's performance in diverse architectural settings

combining the different losses, we compared the results obtained by using ModReduce framework for training a student network against the three underlying state-of-the-art methods used for each type: Hinton's response-based knowledge distillation, SemCKD's feature-based knowledge distillation, and CRD's relation-based knowledge distillation. We created a comprehensive benchmark that combines the experiments conducted by CRD and SemCKD on CIFAR-100 [18], resulting in 15 different combinations of teacher-student architectures. The accuracy metrics for the teacher model, base student model, Hinton model, SemCKD model, CRD model, and the student trained with our "ModReduce" framework are reported in Tables I and II. The student trained using ModReduce surpassed the underlying methods in 10 out of 15 experiments and has an Average Relative Improvement (ARI) up to 48.29%

It is worth noting that in the offline training of different students reported in tables I and II, a weight for Hinton loss was added to both SemCKD and CRD students. This adjustment was made to account for the fact that SemCKD reported their results with Hinton loss included, and CRD reported slight improvement when combining Hinton Loss. Therefore, we re-ran all the experiments for Hinton, SemCKD + Hinton, and CRD + Hinton to ensure that the reported results are accurate and consistent.

5. Results and Discussion

In all the experiments, the 15 teacher-student models combinations shown in tables I and II were used to obtain the conclusions for our questions. To capture the improvement of our model over the existing knowledge distillation techniques, we utilized ARI; a metric that was first introduced by CRD [13] and was later used by SemCKD [12] in reporting their results. ARI provides a measure to test whether, on average, for the set of different architectures, ModReduce improved upon a certain knowledge distillation technique or not.

$$ARI = \frac{1}{M}\sum_{i=1}^{M}\frac{Acc^{i}_{ModReduce} - Acc^{i}_{KD}}{Acc^{i}_{KD} - Acc^{i}_{Stu}} * 100\% \quad (3)$$

The first question was whether the combination of offline losses introduced by Hinton, CRD, and SemCKD into a single loss function could improve upon using each loss function independently. For that, we ran an experiment that performs an aggregated distillation by adding the loss functions of Hinton, SemCKD, and CRD. Figure 2 shows that aggregating the three losses by simply adding the loss functions with their weights improves only upon the Hinton model. At the same time, it has equivalent performance to SemCKD and lower performance than CRD. From this experiment, we can conclude that simply combining the loss functions of the different knowledge distillation techniques does not improve the accuracy of the student model.

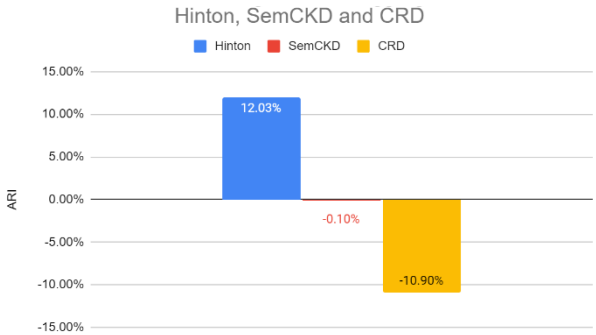


Figure 2. Comparison of Average Relative Improvement (ARI) when combining loss functions from different knowledge types. The results demonstrate that, while simply combining the losses improves over Hinton, it falls short compared to SemCKD and CRD. This highlights the need to find a more refined way for aggregating the different knowledge types.

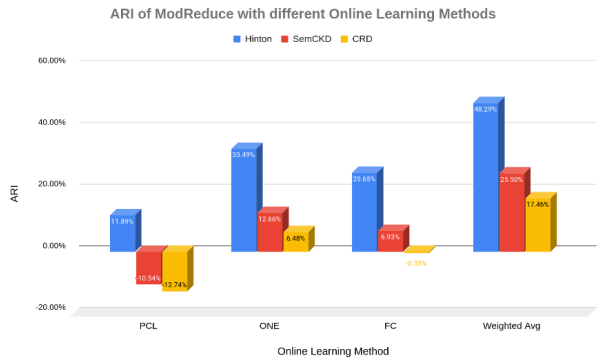


Figure 3. The chart compares the effectiveness of four online learning methods integrated withing the ModReduce framework. The Weighted Average and ONE methods provide the most improvement with a significant increase in ARI over the underlying offline methods while PCL and FC provide limited improvement. Weighted Averaging is selected as the default online learning method in ModReduce given it has the highest ARI over the offline methods used.

The next step was testing whether online aggregation of offline knowledge sources could improve the student's accuracy. We experimented with four aggregation methods to create the teacher logits for the online learning step. Two of them were based upon the Peer Collaborative Learning [10] and On-the-Fly Native Ensembling [11]. The second aggregation method used a fully connected trainable layer to calculate the online teacher logits. Figure 3 shows a graph of the average relative improvement of the different aggregation methods used with ModReduce. ModReduce with ONE and Weighted Averaging (WAvg) have a positive ARI compared to training a student using any of the underlying offline methods. Furthermore, using ModReduce with WAvg has the highest ARIs over all the underlying offline methods; those being 48.2%, 25.50%, and 17.46% over Hinton, SemCKD, and CRD, respectively. Tables I and II show detailed results for the different experiments. ModReduce, with WAvg as the aggregation method for the online learning step, is better in 10 out of the 15 experiments. As for the few experiments in which ModReduce-WAvg does not have the best results, we observed that it is consistently a close second. While analyzing the five experiments in which ModReduce-WAvg is not the best performing, we observed that it is consistently a close second. For instance, in Figure 4a, despite trailing CRD, we notice that ModReduce-WAvg has an accuracy of 69.23%, improving up on SemCKD. On the other hand, Figure 4b shows the opposite case where ModReduce-WAvg is trailing SemCKD. However, with an accuracy of 76.96%, it improves upon the state-of-the-art relational knowledge distillation (CRD).

These results show that aggregating knowledge from different learned-students through online distillation generalizes better than a single offline knowledge distillation technique; Thus, on average, ModReduce-WAvg produces better student models. Moreover, the whole architecture is open for training homogeneous and heterogeneous student models.

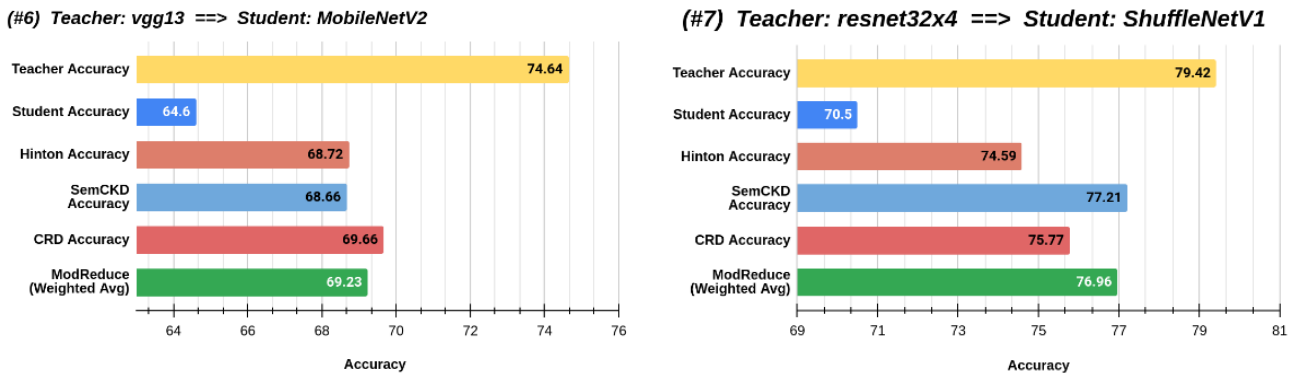
6. Conclusion

In this work, we introduced ModReduce: a novel multi-knowledge distillation with online learning framework. ModReduce aggregates knowledge distilled from three different sources: response-based, feature-based, and relational-based knowledge. This aggregation is performed via online learning between students, which also boosts their performance and enhances their generalization ability. Our experiments proved that using online learning as an aggregation method for different knowledge sources is better than combining the losses in a single student. We also showed that distilling three knowledge types is better than only using one or two types. Our results surpass the state-of-the-art SemCKD and CRD distillation schemes in 10 out of 15 experiments. More specifically, ModReduce outperforms SemCKD in 6 out of 7 experiments and outperforms CRD in 7 out of 11 experiments. Using the Average Relative Improvement (ARI) metric, ModReduce achieved 48.29% improvement over Hinton, 25.5% improvement over SemCKD, and 17.64% over CRD. We believe that ModReduce is the first architecture that introduces distilling three different knowledge sources, leverages a combination of offline and online distillation schemes and allows further experimentation as it inherently supports training homogeneous and heterogeneous student models.

7. Future Work

Knowledge distillation is a prominent field with many research opportunities that could result in better performance and less costly architectures. Even though ModReduce has already achieved results that surpass the state-of-the-art benchmarks, a wide range of potential enhancements can be conducted.

We propose investigating the effect of switching from a synchronous offline-online training scheme to a sequential one (offline followed by online,) a potential enhancement to be tested. We have only replicated the experiments reported in SemCKD and CRD benchmarks in our work. However, other variations of teacher and student model architectures can be tested for further insights. Training student with different architectures, as the three students trained had the same architecture in any given experiment, might also be a good point to explore. Moreover, we encourage the research community to widen the benchmark by experimenting on other network types (since most of the work on knowledge distillation is done on CNNs and image classification tasks). Finally, we propose doing an extensive ablation study to see the effect of the different components of the system on the eventual result. This ablation study can test the effect of changing all variables or parameters one at a time.



(a) VGG13 teacher and MobileNetV2 student

(b) ResNet32x4 teacher and ShuffleNetV1 student

Figure 4. Accuracy comparison of ModReduce-trained student against Hinton, CRD, and SemCKD in cases where ModReduce did not does not achieve the top accuracy. In both scenarios, the ModReduce-trained student remains a close second to the best-performing method, whether it is SemCKD or CRD. These results indicate that while ModReduce may not always achieve the highest accuracy, it consistently provides competitive results, even in challenging scenarios with significant capacity gaps between the teacher and student models. This highlights the robustness and versatility of the ModReduce framework.

References

[1] G. Hinton, O. Vinyals, J. Dean *et al.*, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, vol. 2, no. 7, 2015.

[2] Q. Zhong, L. Ding, L. Shen, J. Liu, B. Du, and D. Tao, "Revisiting knowledge distillation for autoregressive language models," *arXiv preprint arXiv:2402.11890*, 2024.

[3] C. Yang, Z. An, L. Cai, and Y. Xu, "Mutual contrastive learning for visual representation learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 3, 2022, pp. 3045–3053.

[4] J. Rao, L. Ding, S. Qi, M. Fang, Y. Liu, L. Shen, and D. Tao, "Dynamic contrastive distillation for image-text retrieval," *IEEE Transactions on Multimedia*, vol. 25, pp. 8383–8395, 2023.

[5] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, 2021.

[6] R. Müller, S. Kornblith, and G. Hinton, "Subclass distillation," *arXiv preprint arXiv:2002.03936*, 2020.

[7] F. Tung and G. Mori, "Similarity-preserving knowledge distillation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1365–1374.

[8] N. Passalis, M. Tzelepi, and A. Tefas, "Heterogeneous knowledge distillation using information flow modeling," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2339–2348.

[9] S. Kornblith, M. Norouzi, H. Lee, and G. Hinton, "Similarity of neural network representations revisited," in *International Conference on Machine Learning*. PMLR, 2019, pp. 3519–3529.

[10] G. Wu and S. Gong, "Peer collaborative learning for online knowledge distillation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 12, 2021, pp. 10 302–10 310.

[11] X. Zhu, S. Gong *et al.*, "Knowledge distillation by on-the-fly native ensemble," *Advances in neural information processing systems*, vol. 31, 2018.

[12] D. Chen, J.-P. Mei, Y. Zhang, C. Wang, Z. Wang, Y. Feng, and C. Chen, "Cross-layer distillation with semantic calibration," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 8, 2021, pp. 7028–7036.

[13] Y. Tian, D. Krishnan, and P. Isola, "Contrastive representation distillation," *arXiv preprint arXiv:1910.10699*, 2019.

[14] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[15] Y. Liu, J. Cao, B. Li, C. Yuan, W. Hu, Y. Li, and Y. Duan, "Knowledge distillation via instance relationship graph," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7096–7104.

[16] W. Park, D. Kim, Y. Lu, and M. Cho, "Relational knowledge distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3967–3976.

[17] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," *arXiv preprint arXiv:1412.6550*, 2014.

[18] A. Krizhevsky, "Learning multiple layers of features from tiny images," University of Toronto, Tech. Rep., 2009.