Knowledge and Dataset Distillation on MNIST

Elif Ozcan
Computer Engineering Department
Istanbul Technical University
Istanbul, Turkey
ozcane22@itu.edu.tr

Mustafa Aktas
Computer Science Department
Istanbul Technical University
Istanbul, Turkey
aktasm22@itu.edu.tr

***Abstract*—Condensing knowledge from a complex model into a more straightforward one is a technique known as knowledge distillation. It comes from the field of machine learning, where the objective is to build models that can learn from data and predict the future. Making a smaller, less complex model replicates a larger, more complex model to generalize from data is the central concept behind knowledge distillation. Two distinct distillation techniques will be utilized in this project to achieve this goal, and the results will be compared to those of the model trained without distillation as well as each other. The teacher-student model is used in the first technique to reduce the model's complexity, and the training dataset's size is reduced in the second method to reduce both the size of the dataset and the amount of space it takes up during training. Also, our code available on GitHub.**[1]

***Index Terms*—Dataset Distillation, Knowledge Distillation, Response-based Knowledge Distillation, Average Real Images**

I. Problem Statement

A common method for transferring knowledge from a "teacher" model to a "student" model is data distillation. In this procedure, the knowledge produced by the teacher model is reduced into a more digestible form for the student model's training. This method is frequently helpful when the student model is lacking in computational capability or data.

Data distillation often transforms the teacher model's outputs (predictions or characteristics) into a format that the student model can understand. This frequently enables the student model to quickly and efficiently absorb the knowledge of the teacher model.

For an image classification task, for instance, a teacher model can be trained on a huge number of photos. For each image, this model may then produce a class label and a probability distribution. Using this data, a condensed and simplified dataset may be produced for the student model to learn from. The data distillation process is implemented differently depending on how well the instructor model's knowledge is transferred to the student model, and this procedure frequently necessitates a certain level of expertise.

Instead of using the complete training dataset to train the model, the main idea is to use the training data to learn how to learn. This is because not all of the data is pertinent to all of the examples; therefore, eliminating irrelevant data will facilitate training. In some instances, the accuracy of the model can even be improved by deleting some of the unnecessary data. Data distillation is the process of eliminating irrelevant data to facilitate training.

The use of data distillation machine learning is necessary to speed up training and testing instead of ideally for an entire class of models, reduce the amount of data required for storage (Instance Selection), and provide an answer to the scientific question of how much information is contained in the data or how much we can compress it.

Knowledge Distillation and Data Distillation are used in various fields and scenarios:

- Light-weighting Deep Learning Models: The knowledge of a large and complicated model (teacher) is frequently transferred to a lighter model (student) through the process of knowledge distillation. This is helpful for models that frequently run on hardware with little processing power (like mobile or Internet of Things devices).
- Model Compression: It is possible to condense the information contained in a large model into a smaller model using information and data distillation techniques. As a result, the tutor model is smaller and uses less processing power.
- Improving Learning Efficiency: Data distillation can help the student model learn more knowledge with fewer data, increasing learning efficiency. This can be particularly helpful in data-intensive circumstances.
- Better Generalization: Knowledge and data distillation can aid the student model in making more accurate generalizations about a larger range of data. The teacher model's expertise can reflect a wider range of data, which can assist the student model to develop a stronger all-around capacity for learning.
- Noise Reduction: Data distillation can prevent the student model from learning from noisy data by reducing noise. By concentrating on information that is clearer and more pertinent, the teacher model can lessen the effect of noise.
- Secure and Confidential Learning: By allowing the teacher model to distill private information and the student model to learn from this distilled information, data distillation can be utilized to protect data confidentiality

[1]https://github.com/Mstfakts/Knowledge-and-Dataset-Distillation-via-MNIST

where private data sets are employed. This is crucial in industries like healthcare and finance where private and sensitive data are frequently used.

There are mainly two approaches for distillation. These are knowledge distillation for distillate the model and the dataset distillation for the dataset that is used during training. This study, it was aimed to utilize one of each approach to compare the results. For this aim; the "response-based knowledge distillation" and the "average real images" techniques were hired.

II. Literature Survey

A. Core-set or Instance Selection

Core set and instance selection are widely employed concepts within the domains of machine learning and data mining. These techniques aim to diminish the size of datasets while upholding their representative and informative characteristics. Let us delve into each concept individually.

1) Core Set: A core record refers to a subset of the original record that encapsulates the essential information present in the complete record. Its purpose is to achieve a reduced size while retaining the comprehensive and representative nature of the original data. Core sets prove particularly valuable in scenarios where storing, computing, and analyzing entire datasets is either unfeasible or inefficient. The process of constructing a core set involves the selection of a subset of instances that best exemplify the overall distribution and features of the complete dataset. Various methodologies, such as clustering algorithms, optimization techniques, and heuristics, can be employed in creating the core set [5]. The objective is to choose a reduced set of instances that preserves the most crucial information contained within the data.

2) Instance Selection: Instance selection, also referred to as data aggregation or prototypical selection, entails the extraction of a subset of instances from a dataset while maintaining representatives and diminishing redundancy. The ideally chosen instances should encompass the diverse patterns and classes inherent in the original dataset. Methods for instance selection typically involve evaluating the similarity or dissimilarity between instances and employing criteria to determine which instances should be retained and which should be discarded. Similar to core set construction, clustering algorithms, optimization techniques, or instance selection heuristics can be applied. The selected subset of instances ought to effectively summarize the original data distribution, thereby facilitating subsequent analysis or learning tasks in a more expedient and efficient manner [7].

Both cores set construction and instance selection techniques prove beneficial in scenarios where dealing with the entire dataset is hindered by its large size, computational demands, or the presence of redundant or irrelevant instances. By reducing the size of the data while preserving its crucial properties, these methods can enhance the efficiency and efficacy of various machine learning and data mining applications.

3) Coresets for Data-efficient Training of Machine Learning Models : In [6], the paper focuses on the utilization of core sets as a means to enhance the data efficiency of training machine learning models. Coresets are reduced subsets of the original dataset that preserve its representative and informative properties. By constructing a core set that accurately captures the essential information contained in the full dataset, computational resources can be effectively allocated, and training time can be significantly reduced. The authors explore different techniques for constructing core sets, such as clustering algorithms, optimization methods, and heuristics, aiming to select a subset of instances that best represent the overall distribution and characteristics of the original dataset. The experimental results demonstrate that the employment of coresets enables the training of machine learning models with reduced data, without significant loss in performance. This paper contributes to the development of data-efficient approaches in machine learning and provides insights into the potential applications and benefits of using core sets for model training.

B. Hyperparameter Optimization

Hyperparameter optimization and dataset distillation are two separate concepts in machine learning. Hyperparameter optimization involves finding the best values for parameters that control a model's behavior and performance [9]. On the other hand, dataset distillation aims to reduce dataset size while maintaining its key characteristics. While these two concepts are typically distinct, it is possible to view dataset distillation as a form of hyperparameter optimization in certain scenarios.

By viewing dataset distillation as a hyperparameter optimization problem, one can approach it with similar methodologies and techniques. The optimization process can involve searching for the best hyperparameters that maximize the preservation of important information while minimizing the dataset size. Techniques such as grid search, random search, Bayesian optimization, or genetic algorithms can be adapted for this purpose.

In summary, although dataset distillation and hyperparameter optimization are distinct concepts, one can interpret dataset distillation as a form of hyperparameter optimization by considering the hyperparameters involved in the distillation process and applying optimization techniques to select the best hyperparameter settings for achieving the desired dataset reduction while maintaining important characteristics [8].

1) Dataset Distillation: Dataset distillation aims to reduce the size of a given dataset while preserving its essential characteristics.

In [8], the authors propose a two-stage approach for dataset distillation, involving an initial clustering step followed by an instance selection step. In the first stage, the authors cluster the data instances based on their similarities using a clustering

algorithm. In the second stage, they select a representative subset of instances from each cluster, using an optimization algorithm that seeks to maximize the information retained in the selected subset. The authors evaluate their dataset distillation technique on several standard image classification datasets and demonstrate that their approach can significantly reduce the size of the dataset while maintaining comparable or even improved performance when training a deep neural network on the distilled dataset. The paper's contributions lie in proposing a new technique for dataset distillation that involves clustering and instance selection and demonstrating its effectiveness in reducing the size of a dataset while preserving its essential characteristics. The proposed technique can have significant practical applications in scenarios with limited computational resources or large datasets, where reducing the dataset size can bring computational and storage advantages.

C. Knowledge Distillation

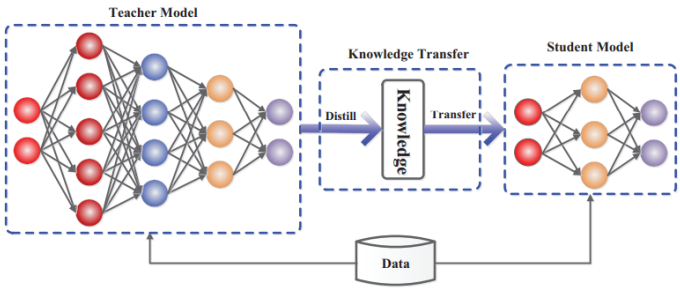

Fig. 1. The generic teacher-student framework for knowledge distillation. [1]

Knowledge Distillation is a process that aims to transfer the knowledge of a larger and more complex teacher model to a smaller and faster working student model. Due to the performance of the instructor model being leveraged in this process, the student model operates more quickly and effectively. Mainly to minimize the size and computing requirements of deep learning models is knowledge distillation. In this approach, the performance of quick and tiny models can be comparable to that of large ones.

According to the [1], there are 3 main types of knowledge distillation. These are "Response-Based", "Feature-Based", and "Relation-Based".

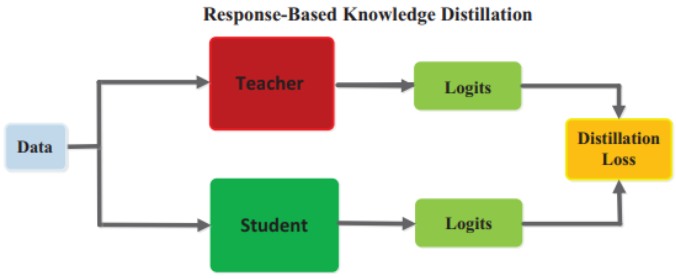

Fig. 2. The generic response-based knowledge distillation. [1]

1) Response Based Knowledge Distillation: Response-Based Knowledge Distillation focuses on the final output layer of the teacher model. With this method, the student model is taught to imitate the teacher model's predictions. This can be accomplished by employing a loss function, or distillation loss, that measures the disparity between the logits (before the final output layer) of the student and instructor models.

In this process, the distortion loss decreases over time, allowing the student model to become more accurate in making predictions similar to the teachers. Therefore, Response-Based Knowledge Distillation helps the student model to learn to mimic the performance of the teacher model. However, this approach does not guarantee that the student model fully grasps all the features or knowledge of the teacher model, and this may constitute its limitations in some cases.

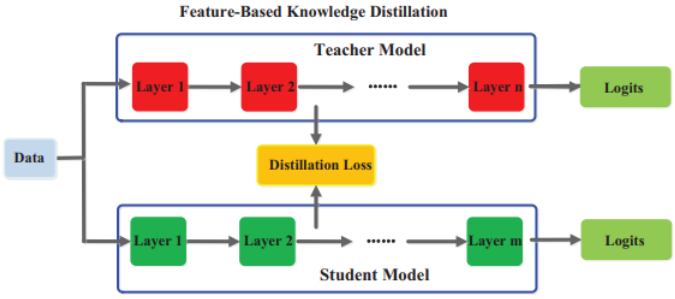

Fig. 3. The generic feature-based knowledge distillation. [1]

2) Feature Based Knowledge Distillation: When learning multi-layer feature representations with increasing abstraction, deep neural networks excel. In its intermediate layers, a trained teacher model also captures data, which is crucial for deep neural networks. The intermediate layers pick up on the differences between particular features, and a student model can be trained using these features.

The aim of the Feature-Based Knowledge Distillation approach is to ensure that the student model learns the same feature activations as the teacher model. This is achieved by minimizing the difference between the feature activations of the teacher and student models.

This approach ensures that the student model learns to emulate the output layer of the teacher model while at the same time learning the feature outputs of the teacher model. This enables the student model to have a stronger and more generalized learning ability and thus helps the student model to effectively mimic the teacher model's performance.

3) Relation Based Knowledge Distillation: Both response-based and feature-based knowledge draw on the results of particular tutor model layers. The relationships between several layers or data instances are expanded through Relation-Based Knowledge Distillation (RBKD). To evaluate the connections between distinct feature maps, a solution

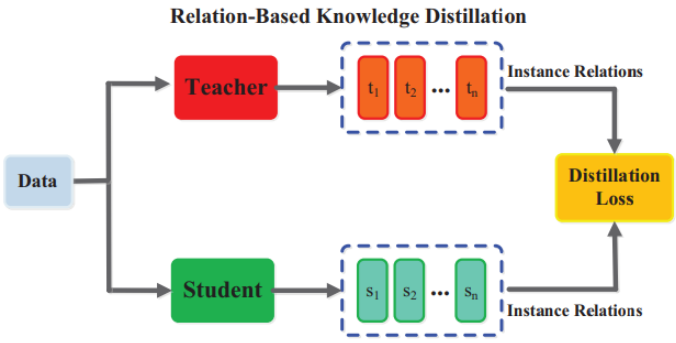

Fig. 4. The generic instance relation-based knowledge distillation. [1]

process flow (FSP) is employed, which is described by the Gram matrix between two layers.

The FSP matrix summarizes the connections between the feature map pairs. The inner products of the features of the two layers are used to calculate this. Correlations between feature maps are employed as the distilled information in a process called singular value decomposition.

The link between feature maps, graphs, similarity matrices, feature embeddings, and probability distributions based on feature representations can be summed up in this way. The diagram below shows how the paradigm works.

This method gives the student model the ability to comprehend the connections between the tutor model's outputs, which enhances learning ability and performance. However, compared to other knowledge distillation techniques, this approach's implementation can frequently be more difficult.

D. Generative Models

A technique of knowledge distillation known as "knowledge distillation with generative models" frequently employs generative models with the capacity to sample and rebuild data. In this type, a teacher model, typically a Generative Adversarial Network (GAN) or a Variational Autoencoder (VAE), generates data that the student model learns [2].

The instructor model creates fresh data that tries to replicate the distribution of the real data. The student model is trained using this fresh data. This strategy can assist the student model in learning more diversified and all-encompassing data samples, which strengthens and expands the student model's capacity for general learning.

The student model picks up on how to replicate the teacher model's outputs while also picking up on how the teacher model distributes the data. This facilitates more effective knowledge transfer from the teacher model to the student model.

However, because generative models are sometimes intricate models that are challenging to train and because it can be challenging for these models to accurately transfer knowledge to the student model, this strategy can frequently be more difficult to apply than other knowledge distillation methods.

III. Methods

A. Dataset

In the project, we used the MNIST dataset, which is frequently used in similar studies. MNIST consists of images of handwritten digits. The images are gray-scale, which means the pixel values are between 0-255. It contains 28x28 images from 10 different classes with 60,000 train and 10000 test data. The MNIST data is a subset of the much larger NIST.

B. Distillation Methods

1) Response-based Knowledge Distillation: Response based knowledge distillation method is based on transferring the knowledge of a large model, called the teacher model, to the student model and increasing the success of the student model. Due to the size of the student model, the training time and complexity is considerably smaller compared to the teacher model.

When training the student model, it uses information from the final output layer of the teacher model. In this way, the knowledge of the teacher model is transferred to the student model. In our project, we will first train a large teacher model and get the performance of this model. By large model, we mean a deep neural network with convolution filters and multiple fully-connected hidden layers. A smaller student model is then trained with the original data labels for comparison. Finally, the student model is trained using the outputs of the teacher model, where knowledge distillation is performed, as labels and their performance is compared.

For comparison, we used the accuracy of models, training times, and size of models.

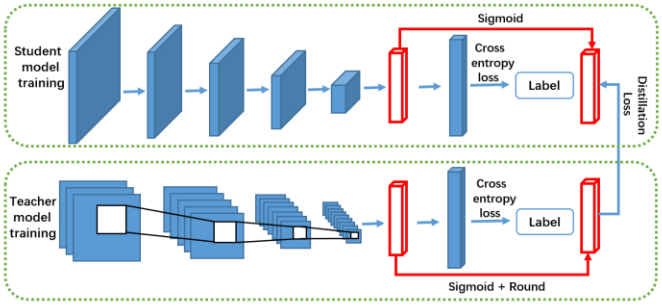

Fig. 5. Training process of teacher model and student model. [3]

2) Average Real Images: Average real images are a data distillation method. Data distillation aims to reduce the number of images used in training, thus reducing training time and model size. Since data distillation does not use the raw data, but creates new samples from the data, data, where privacy is important, can also be used in training.

In the average real images method, new images are created by taking the pixel averages of the data on a class basis and these images are used for training [4]. For example, after the MNIST data is grouped according to classes, for each pixel, the pixel at the same position in the whole data is averaged and the image representing this class is created. In our project, the deep neural network with the same features was first trained with all the data and then trained with different numbers of data such as 10, 100, and 500 from each class, and their performances were compared. Accuracy, training time, and model size were used to compare the performances.

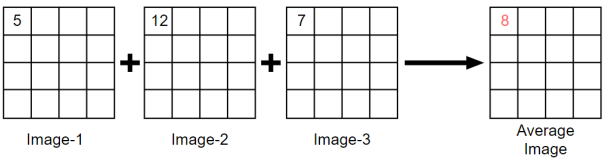

Fig. 6. The process of averaging the images.

IV. Results

In this project, we looked into the dataset and knowledge distillation and compared the models in the aspect of accuracy, number of misclassified samples, model size, and training time. For dataset distillation, we used the average of real images in the training set and used the averages for training the distilled model, for knowledge distillation we used the teacher-student model approach.

The teacher model, also the big model we used for comparing the effectiveness of distillation techniques, has 3.5 as Softmax Temperature, Adam optimizer, Sparse Categorical Crossentropy for loss function, batch size of 32, and trained for 1 epoch. The student model also has 3.5 as Softmax Temperature, Adam optimizer, batch size of 32, and Categorical Crossentropy for loss function, cross-validation set to 3 and trained for 3 epochs just like the small model but with the teacher model's weights. In Table-1 you can see the results of these 3 models. As seen although the small model and student model have the same parameters the student model performed better.

For dataset distillation, we used the average of the training set and tried 20, 100, 500, and 2500 average samples for training. The model has the same parameters as the teacher model for comparison. As seen in Table-2 as the number of samples increased the accuracy also increased. Also for the sake of comparing the effectiveness of averaging images in terms of dataset selection, we selected the same number of samples and compare the models.

TABLE I
Model Performances and Features

Model	Accuracy	Misclassified	Model Size	Training Time
Teacher Model	0.9777	222	157	581
Small Model	0.8211	732	81	3
Student Model	0.9302	545	81	4

V. Discussion and Conclusions

In this paper, we have presented a comprehensive study on both dataset and knowledge distillation techniques to enhance the training and performance of deep learning models. By distilling knowledge from a large, high-quality dataset, we have shown that dataset distillation enables the creation of compact subsets that capture the essential information necessary for training deep learning models. This reduction in dataset size offers several practical advantages, including reduced computational requirements, faster training times, and improved generalization capabilities.

Furthermore, we have explored the integration of knowledge distillation, where a teacher model is used to transfer its knowledge to a student model. By leveraging the rich knowledge encoded in pre-trained models, we can enhance the learning process of the student model and improve its performance. The distillation process helps the student model to focus on the most relevant information and overcome challenges such as overfitting or limited labeled data availability.

The results obtained from our experiments showed that with knowledge distillation the student model can perform just like the teacher model with less training time and model size. So for example, if you were to use a deep learning model in a small device using the student model would be more practical. The dataset distillation method we used did not perform as well as the teacher model but we could see that it still performed better in comparison to the model with the same number of samples.

For future work trying different dataset distillation methods and also trying the dataset and knowledge distillation can be used together.

In conclusion, our work introduces dataset distillation as a valuable technique for addressing the challenges of training deep-learning models with limited resources. By distilling the knowledge from large datasets into compact subsets, we can overcome limitations in storage, computational power, and data availability without compromising model performance. As for the knowledge distillation of large, complex models, we enable the creation of smaller, more efficient models that can rival or even surpass their larger counterparts. The knowledge transfer process not only enhances the student models'

TABLE II
MODEL PERFORMANCES AND FEATURES

Model	Number of Samples In Train Data	Accuracy	Misclassified	Model Size	Training Time
Teacher Model	60.000	0.9777	222	157	581
Average Real Images	200	0.2406	7594	159	11
Average Real Images	1000	0.4463	5537	159	18
Average Real Images	5000	0.6677	3323	159	53
Average Real Images	25000	0.9358	642	159	234
Selection	200	0.1009	8991	159	11
Selection	1000	0.4165	5835	159	17
Selection	5000	0.8192	1807	159	53
Selection	25000	0.9600	400	160	230

accuracy and generalization but also reduces computational requirements, making them more suitable for deployment in resource-constrained environments.

REFERENCES

[1] Gou, J., Yu, B., Maybank, S. J.; Tao, D. (2021). Knowledge distillation: A survey. International Journal of Computer Vision, 129(6), 1789–1819. https://doi.org/10.1007/s11263-021-01453-z

[2] Yim, J., Joo, D., Bae, J., Kim, J. (2017). A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4133-4141.

[3] H. Zhai, S. Lai, H. Jin, X. Qian and T. Mei, "Deep Transfer Hashing for Image Retrieval," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 2, pp. 742-753, Feb. 2021, doi: 10.1109/TCSVT.2020.2991171.

[4] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, Alexei A. Efros. (2020). Dataset Distillation.

[5] Ruonan Yu, Songhua Liu, Xinchao Wang. (2023). Dataset Distillation: A Comprehensive Review.

[6] Baharan Mirzasoleiman, Jeff Bilmes, Jure Leskovec. (2020). Coresets for Data-efficient Training of Machine Learning Models.

[7] Olvera-López, J.A., Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F. et al. A review of instance selection methods. Artif Intell Rev 34, 133–143 (2010). https://doi.org/10.1007/s10462-010-9165-y

[8] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, Alexei A. Efros. (2020). Dataset Distillation.

[9] Maclaurin, D., Duvenaud, D., Adams, R. (2015). Gradient-based Hyperparameter Optimization through Reversible Learning. In Proceedings of the 32nd International Conference on Machine Learning (pp. 2113–2122). PMLR.