Institut für Technische Informatik
Chair for Embedded Systems

Latent Space Knowledge Distillation under Heterogeneous Data

Masterarbeit
von

B.Sc.
Matthias Schmitt

am Karlsruher Institut für Technologie (KIT)
Fakultät für Informatik
Institut für Technische Informatik (ITEC)
Chair for Embedded Systems (CES)

Erstgutachter:	Prof. Dr. Joerg Henkel
Zweitgutachter:	Prof. Dr. Wolfgang Karl
Betreuer:	M.Sc. Kilian Pfeiffer, M.Sc. Martin Rapp

Tag der Anmeldung:	01.06.2021
Tag der Abgabe:	01.12.2021

Chair for Embedded Systems

Erklärung

Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

Die verwendeten Quellen und Hilfsmittel sind im Literaturverzeichnis vollständig aufgeführt.

Karlsruhe, den 01.12.2021

Matthias Schmitt

Zusammenfassung

Während sich der Einsatz von immer rechenintensiveren künstlichen neuronalen Netzen in vielen Disziplinen durchsetzt, gibt es zeitgleich das Bestreben diese wegweisende Technologie direkt auf Endgeräten mit begrenzten Ressourcen einzusetzen. Föderiertes Maschinelles Lernen beschreibt ein Szenario in dem eine große Anzahl von Teilnehmern mit begrenzter Rechenkapazität gemeinsam ein neuronales Netz trainieren. Lokales Trainieren auf den eigenen Daten und anschließender Austausch der gelernten Netzwerkparameter erlaubt es dem Kollektiv die individuell beschränkte Rechenkapazität zu bündeln, während die Geheimhaltung der privaten Daten gewährt bleibt. Dieses Szenario stellt die zur Konsensbildung eingesetzten Algorithmen vor besondere Herausforderungen, da jede:r Teilnehmer:in über eine unterschiedliche Datenverteilung und meist nur geringe Datenmengen verfügt.

Die vorliegende Arbeit beschreibt eine Technik mit der den Herausforderungen der Knappheit und der Heterogenität der Daten begegnet werden kann. Sie macht sich die Methode der Knowledge Distillation zunutze, die es erlaubt, ein kleineres Netz (Schüler) unter Anleitung eines großen neuronalen Netzes (Lehrer) zu trainieren und die Entscheidungsfunktion des rechenintensiven Lehrers, das "Destillat", im effizienteren Schüler nachzubilden. Besonders geeignet ist diese Technik für die Kooperation im föderalen Maschinellen Lernen, da statt der umfangreichen Netzwerkparameter lediglich die Ausgaben der Modelle ausgetauscht werden müssen. In der hier vorgestellten Erweiterung wird nicht die finalen Klassenwahrscheinlichkeiten ausgetauscht, sondern eine interne Repräsentation der Eingabe. Dadurch, dass der Wissenstransfer im Repräsentationsraum (latent space) stattfindet, sorgt Latent Space Knowledge Distillation für einen effizienten Wissensaustausch bei Datenheterogenität und erhält gleichzeitig eine gewisse Anpassungsfähigkeit des Schülernetzes.

Die Kompatibilität der internen Repräsentationen von Schüler und Lehrer wird durch das Einfügen einer zusätzlichen voll-vernetzten linearen Schicht in beide Modelle sicher gestellt. Im Destillationsprozess wird die intern aufgebauten Wissenrepräsentation des Lehrers, durch das Minimieren des Abstands der Schülerrepräsentation zur der des Lehrers, vom Schüler imitiert. Die Effizienz der vorgestellten Erweiterung von Knowledge Distillation wird in Schüler-Lehrer-Szenarien mit unterschiedlich vielen Daten und verschiedenen Heterogenitätsgraden evaluiert. Außerdem befasst sich die Arbeit mit dem Aufbau des Repräsentationsraums und führt eine zusätzliche Methode zur robusteren Destillation von Repräsentationen ein. In den Experimenten zeigt sich, dass Latent Space Knowledge Distillation für den Einsatz in der Realität nachempfundenen heterogenen Szenarien geeignet ist und bessere Ergebnisse erzielt als alternative Destillationsansätze.

Abstract

This thesis describes a technique that deals with the problems of heterogeneous and scarce data that occur in federated scenarios where diverse participants collaboratively train deep neural networks. Knowledge distillation is a model compression method that reproduces the decision function of a large neural model (teacher) in a smaller neural net (student). During training, the student is guided by the output of the teacher and mimics the output. Distillation is particularly advantageous for cooperation in federated learning. Instead of the network's parameters, just the outputs of the models need to be exchanged. Rather than communicating the model's final output, the extension to knowledge distillation proposed in this thesis transmits an internal representation of the input. Latent Space Knowledge Distillation achieves efficient knowledge transfer under heterogeneity by transferring the knowledge in the representation space and retains the student's capacity to adjust to the local distribution.

A fully-connected linear layer is appended to both the teacher's and the student's model to create compatible internal representations. By minimizing the distance between the student's representation and the representation of the teacher, the student imitates the internal abstractions built by the teacher. The efficiency of the proposed extension to knowledge distillation is evaluated in a student-teacher scenario with a varying amount of data and data heterogeneity. Additionally, the thesis is concerned with the configuration of the latent space and introduces an augmentation procedure to make latent space distillation more robust. The experiments undertaken for this thesis show that latent space distillation is better suited for distillation in heterogeneous environments, and it surpasses alternative extensions to knowledge distillation.

Contents

1 Introduction 3

1.1 Motivational Example 5

1.2 Contribution 7

2 Fundamentals 9

2.1 Knowledge Distillation 9

2.2 Latent Space and Visualization 10

2.3 Dataset and Heterogeneity 12

3 Related Work 15

3.1 Knowledge Distillation 15

3.2 Federated Machine Learning 16

4 Latent Space Knowledge Distillation 19

4.1 Problem Definition 19

4.2 Distillation in the Latent Space 20

4.3 Scheme for Heterogeneous Distillation 22

4.4 Latent Space Augmentation 23

5 Experimental Evaluation 25

5.1 Experimental Setup 25

5.2 Data Scarcity and Heterogeneity 26

5.3 Latent Space Dimensionality 28

5.4 Feature Augmentation 29

5.5 Comparison 31

5.6 Summary 32

6 Conclusion 35

6.1 Outlook 35

List of tables 37

List of figures 38

Bibliography 40

Chapter 1

Introduction

> Sharing knowledge is the most fundamental act of friendship. Because it is a way you can give something without loosing something.
>
> *Richard Stallman*

The ubiquity of deep neural network architectures in state-of-the-art algorithms for e.g., speech recognition [Cha+16; ZXX18], machine translation [Vas+17], protein folding [Jum+21] and many other fields is the result of important scientific progress in both theoretical concepts and the capability of the underlying hardware.

On the hardware side of machine learning research, GPUs and specialized chips for neural computation like Googles TPUs improved the training efficiency and paved the way for the triumph of artificial neural networks. As the hardware for neural computation gets more efficient, algorithms previously only feasible to run in large computer clusters can be moved to the mobile computing edge. Nowadays, recent middle to high-class smartphones are equipped with neural coprocessors, and support for the specialized hardware in mobile contexts is coming to popular machine learning frameworks.

While the possibility of moving neural computation to the edge is arriving, new techniques in the field of machine learning often require more computational power and the deployment of state-of-the-art models in mobile scenarios remains an important research direction. Several techniques to bring the inference resource requirements of neural networks down like weight quantization [Hub+17], pruning non-essential parts [RSN20] and light weight architectures [How+17; San+18] have been proposed. A model compression technique net called *knowledge distillation* that is compatible with these approaches that improve one specific neural net or architecture. Therefore, this technique has attracted a lot of interest. Instead of training small networks from scratch, in knowledge distillation, a smaller more efficient model is trained under the guidance of an accurate but cumbersome to deploy model. The smaller model is called the *student* because it is trained to imitate the output of the bigger model, the *teacher*. This way, student receives additional information about the decision made by the teacher on the training examples. This improves the student's performance compared to training solely by itself by conveying class relations discovered by the teacher. Modifications to the vanilla single teacher-single student knowledge distillation algorithm have been proposed e.g., ensembles of teachers have been shown to further improve the quality of the student. Knowledge distillation helps the deployment of cumbersome models by transferring their expressiveness to smaller more efficient models without loosing performance.

Novel personalized machine learning application on mobile device, made possible by the availability of neural computing on smartphone and edge devices, will pose different challenges to the underlying neural algorithms. Personalized models for spelling correction and grammar prediction, recommendation systems or medical advice demand for strict privacy of the users. The training data for these new applications is real-world data generated by the user on the device. The data is inherently distributed, more relevant than generally available proxy data and highly privacy sensitive. A framework that fits these requirement is *federated learning* (FL). There, the participating clients are assumed to be constrained edge devices like mobile phones and the data on these devices is considered private and not to be disclosed to a server nor to other clients. The clients individually train the global model on their private data. Consensus on the global model is reached through the periodical exchange and aggregation of model parameters or gradients at central servers rather than exchanging their raw data [Ita+21]. With the combined computational resources and data of all clients the federated approach presents a viable alternative to models deployed in the cloud.

Emancipating from the model compression aspect, distillation has been shown to also be beneficial in scenarios where no pre-trained teacher is available and multiple students share knowledge while learning simultaneously [Zha+18a]. While convolutional neural networks make the spatial structure of data available and recurrent neural networks bridge the temporal dimension of data, knowledge distillation can embrace the distributed nature of data and knowledge by broadening the focus of machine learning to interconnected models rather than standalone models. This realization makes distillation a good option for a federated learning environment. Federated distillation (FD) communicates the output of the client's models instead of the parameters and utilizes these outputs for transfering the client's knowledge into the global models. Through knowledge distillation on the aggregated outputs consensus on the global model is reached and later transferred back to the clients. While the exchange of model outputs instead of model parameters saves communication bandwidth, more importantly it allows for model heterogeneity across the participating clients. Allowing the collaboration of clients with heterogeneous hardware constraints is important, as hardware fragmentation is a common situation in the mobile and IoT setting.

In a realistic federated scenario with fragmentation in hardware and models, the diverse clients will also have their own distributions of data. The clients try to solve related but personalized problems with an individual distribution of data. This fact presents a struggle for federated approaches. The assumption of independent identically distributed (iid) data is critical for many theoretical results of statistics and machine learning. In statistics, random variables that are independently drawn from the same probability distribution often simplify mathematical formulations. Thus, they allow for strong results such as the central limit theorem. In machine learning, the iid assumption is made with regard to the data points in the dataset which are assumed to originate from the same memory-less generative process and distributed uniformly. The assumption also guaranties the correctness of using stochastic gradient decent as the error landscape remains smooth over mini batches. Concerns have been raised that because of subjective class assignment (e.g., what counts as spam for the individual clients) or unfeasible complexity of an all distributions encompassing model given the hardware constraint, it might not be possible to train a model to fit all client's distributions [SMS20].

A solution in the context of federated distillation is to view these heterogeneous data distributions as multiple tasks in the same domain. Instead of agreeing on a single global model each client trains a local model that fits the local data distribution. Benefiting from the computation and data of other clients by training local models and relying on knowledge distillation to align those models leaves the question of how to efficiently and collaboratively distill knowledge under heterogeneous data. To build the foundations of effective federated distillation, this thesis works on understanding and improving the efficiency of knowledge distillation under heterogeneous data i.e., non independent identically distributed.

1.1 Motivational Example

Two preliminary experiments introduce and motivate the research question this work addresses. The focus in this introduction lies on the conclusions and insights that can be drawn from the experiments. A rigorous and detailed experimental setup (the same as for the experiments in later sections) is given in section 5.1.

Both experiments address the classical image classification task on the CIFAR10 [Kri09] dataset and utilize a teacher model to help the training of different students via knowledge distillation. The teacher's architecture consists of several convolutional layers followed by a fully connected representation layer of size 500 and an output layer to the 10 classes of the dataset (Figure 1.1). The student model uses the same architecture but with fewer layers and kernels in the convolutional part of the network and is more computational efficient. The students receive the output of the teacher (alternatively the output after the representation layer) for every training image and in addition to correctly classifying the image the student is trained to match the output of the teacher.

After training for 200 epochs on the full dataset, the teacher model reaches a top-1 accuracy of 0.88. In the same scenario the student model would reach a top-1 accuracy of 0.80 showing the capacity advantage of the more expensive teacher model over the lightweight student model. To simulate the condition of real world federated learning in the experiments, the student networks are trained on a subset of 4096 images from CIFAR10 and the class distribution is subject to a synthetically introduced heterogeneity with the Dirichlet distribution.

The first experiment highlights the difficulties of knowledge distillation under heterogeneous data. In one scenario, the class distribution of the training data is uniform and in the other the data is synthetically made heterogeneous resulting in a biased non-uniform class distribution. In each scenario a student is trained to mimic the output of the teacher model on the training data. For comparison, another student model is trained regularly without knowledge distillation. To see how the students accommodate to the non-iid training data and how severe performance degrades on the teachers task, two test results are reported. The first test result is based on the uniform distributed dataset. The second

data	variant	accuracy on iid	accuracy on non-iid
iid	regular training	0.592	
iid	knowledge distillation	0.590	
non-iid	regular training	0.549	0.686
non-iid	knowledge distillation	0.574	0.603

Table 1.1: Experimental results showing the difficulties of knowledge distillation with data heterogeneity

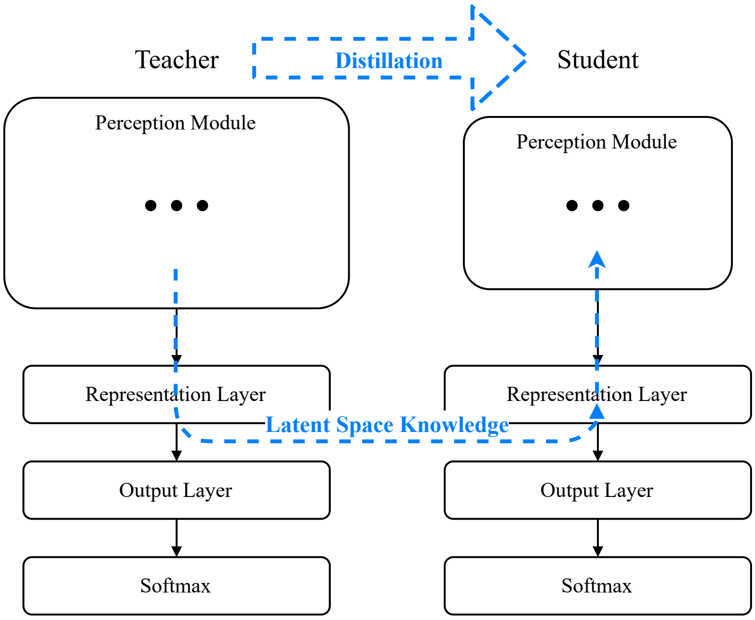


Figure 1.1: Fundamental concept of Latent Space Distillation

test result is based on the test data sampled in the way that is has the same distribution as the training data.

The results of the first experiment are reported in Table 1.1. In the normal (iid) scenario, both methods reach a similar accuracy on the test set. The students in the non-iid scenario performs better on the non-iid test data than the iid students on the iid test data. The reason is that the skewed distribution in the non-iid scenario allows to disregard the rarer classes and focus on the frequent classes. In the non-iid scenario, the student trained with knowledge distillation performs better on the uniform test data. It indicates that by imitating the teacher information about the unbiased distribution was transferred. More importantly, distillation performs much worse on the test data that follows distribution of the training data. The fact that knowledge distillation does not improve the model for the student with heterogeneous data is an obstacle for adoption of knowledge distillation in federated scenarios. The client has no incentive to participate in the federated learning cluster when a model with input from the collective performs worse than a model trained in a regular way.

The research idea of this thesis is that in a non-iid scenario the student can be improved by distilling the information of an intermediate teacher layer. Figure 1.1 shows an overview of the model architecture; the information is extracted after the teacher's representation layer. By following the teacher at an internal level rather than the output, the students internal structure should follow the teacher more closely. The distilled representation should be more useful for a client who is interested in their own task (with heterogeneous class distribution) as the last layer can adapt to the distribution of the classes while profiting from the rich representation that the teacher model learned on the larger dataset.

The second experiment tests this intuition by comparing regular "output" distillation with the effects of distilling the teacher's knowledge from an intermediate layer. In a pre-training step the students distill the teacher under ideal conditions i.e., on the complete training set of the teacher. One student is trained to mimic the output (output layer) of the teacher and another to mimic the teacher's intermediate representation (representation layer). After the students have distilled the teacher their parameters up to representation

data	variant	accuracy on iid	accuracy on non-iid
non-iid	pretrained on output	0.763	0.834
non-iid	pretrained on representation	**0.782**	**0.848**

Table 1.2: Representation distillation on the full dataset

layer are fixed and only the output layer is trained on the non-iid training data.

The results in Table 1.2 show that both students clearly outperform the students of the first experiment. The students of the previous experiments seem to have had trouble distilling a helpful representation under scarce and heterogeneous data. The student who distilled from the internal representation of the teacher managed to create a better representation in the frozen layers and performs better on both the non-iid test data as well as the uniform test data. While this experiment not represent the federated learning scenario exactly, it highlights that distilling the knowledge from an intermediate representation rather than from the final output improves the learned representation when faced with heterogeneity in the training data.

1.2 Contribution

Knowledge distillation is an interesting addition to the federated learning concept because it allows diverse clients with similar tasks but heterogeneous model architectures to collaborate. But — like many algorithms — it suffers from shortcomings when the data is heterogeneous. The motivational examples suggest that using knowledge distillation not only on the output layer but also on an intermediate layer allows the students to learn more robust representations while keeping capacity free to adapt to their individual task.

This thesis explores the effects of data heterogeneity on the efficacy of knowledge distillation and shows that students can be improved by learning from internal representations of the teacher instead of the usual final output (*latent space distillation*). The output of a fully connected layer after the perception part of the model and before the output layer acts as a target for the student to learn the latent space manifold of the teacher. By distilling richer representations and thus building overall better models with latent space distillation, it lays a ground stone for improving the representations of federated distillation algorithms which allow clients to benefit from the data and compute power of other clients without compromising on privacy.

In Chapter 2 the concept present in the title “Knowledge Distillation”, “Latent Space” and “Heterogeneous Data” are explained and formalized. Chapter 3 provides an overview of related work in the field of knowledge distillation and federated machine learning. In the Chapters 4 and 5 the theory of distilling latent space information is explained and evaluated in multiple experiments. Chapter 6 summarizes the findings of this thesis and gives an overview of future research directions.

Chapter 2

Fundamentals

This chapter introduces the significant concepts of this thesis. The essential concept of *knowledge distillation* is a model compression technique where the training of a student model is improved by providing it with knowledge of a more capable but difficult to deploy teacher model. The distillation process allows a high versatility when designing models and is a cornerstone for collaborative learning of heterogeneous clients. The *latent space* is a concept that describes the internal embeddings of neural nets that, with increasing depth, formulate more abstract representations. Several examples highlight the abstractions created by neural models and their latent spaces, and an understanding of the latent space provides a starting point for a more efficient distillation method that works under non-iid data. The last concept, *data heterogeneity*, is synthetically applied to achieve realistic scenarios of federated where diverse clients each have their related tasks but with a different class distribution.

2.1 Knowledge Distillation

At the root of knowledge distillation lies a question of model compression, namely how to compress the function that is learned by a complex model into a much smaller and faster model that has comparable performance [BCN06]. Because deep neural nets are very good function approximators they were used to imitate the output of the target model. In the context of neural nets Ba et al. introduced the term *teacher* for the pre-trained complex model and *student* for the smaller more efficient model into which the knowledge of the "cumbersome" teacher model gets compressed. Hinton et al. popularized the concept under the name knowledge distillation or just distillation [BC14; HVD15].

This thesis builds upon the standard knowledge distillation framework formulated by Hinton et al. that uses so-called *soft targets* as the source of teacher knowledge and is categorized as "response-based" knowledge in an overview over different knowledge distillation approaches by Gou et al. [Gou+21]. To see how the teacher's knowledge can help a student learn the output function of the teacher, imagine an input $\mathbf{x}$ for which the teacher probabilities reveal that the image lies close to the decision boundary, such information will help the student learn the same decision boundary as the teacher.

Formally, given an input $\mathbf{x}$, the teacher network T produces K class-dependent scores also called logits i.e., the output of the last fully connected layer of the deep neural $s^T(\mathbf{x}) = [s_1^T(\mathbf{x}), s_2^T(\mathbf{x}), ..., s_K^T(\mathbf{x})]$. The logits are converted into probabilities by pointwise application of the softmax function (Equation 2.1).

$$p_i(\mathbf{x}) = \frac{\exp(s_i(\mathbf{x}))}{\sum_j \exp(s_j(\mathbf{x}))} \tag{2.1}$$

The soft targets $p(\mathbf{z}, \tau)$ are a temperature scaled softmax of the logits $\mathbf{z} = s^T(\mathbf{x})$ of the teacher model

$$p(\mathbf{z}, \tau)_i = \frac{\exp(\mathbf{z}_i/\tau)}{\sum_j \exp(\mathbf{z}_j/\tau)} \tag{2.2}$$

where τ is the temperature hyperparameter. A temperature of 1 gives the standard softmax function and with larger τ more emphasis is put on the smaller values of the class distribution. On the other side, a temperature close to 0 produces one-hot vectors $\mathbf{y}_i = \mathbb{I}_{i==k}$ where all probability weight is concentrated on the predicted class k. Since this would remove information about how the teacher model's decision and its internal structure, it is common to restrict the temperature $\tau \geq 1$.

In the standard "offline" distillation scheme first, the teacher is trained till convergence on a training set, then the student is trained on a transfer dataset, which could be the same dataset the teacher trained on or a different one. During the training of the student, a distillation loss between the soft targets of the teacher and the soft targets of the student (with the same temperature) guides the student by revealing information about the relative class associations.

Most of the early knowledge distillation literature uses the Kullback-Leibler (KL) divergence $\mathbb{E}_{\mathbf{x}\sim X}[\log(p(\mathbf{s}^S, \tau)) - \log(p(\mathbf{s}^T, \tau))]$ to minimize the distance between the soft target distributions in the knowledge distillation loss. Some publications successfully use the mean squared error (MSE) $\mathbb{E}_{\mathbf{x}\sim X}[(p(\mathbf{z}^S, \tau) - p(\mathbf{z}^T, \tau))^2]$ an intuitive KD loss function to match the soft targets. In a novel publication, Kim et al. empirically show that $\mathcal{L}_{MSE}$ improves the performance and that the penultimate layer representations follow the teacher closer than with $\mathcal{L}_{KL}$ [Kim+21].

When correct labels for the transfer dataset are available the standard cross-entropy between the one-hot encoded class label and the student output is used in addition to the distillation loss. Often a linear combination between both losses is used but in this thesis, the distillation loss is scaled

$$\mathcal{L} = \mathcal{L}_{CE}(p(\mathbf{z}^S, 1), \mathbf{y}) + \alpha\, \mathcal{L}_{KD}(p(\mathbf{z}^T, \tau), p(\mathbf{z}^S, \tau)) \tag{2.3}$$

where $\mathbf{y}$ is the target label, $\mathbf{z}^T$ and $\mathbf{z}^S$ are the logits of the teacher and the student and α is a scaling parameter. This formulation is equivalent when the learning rate is adjusted.

Since the introduction and popularization of the knowledge distillation paradigm in 2015 many modifications and additions have been discussed. A review of all variants is beyond the scope of this thesis and the reader is referred to a comprehensive study of Gou et al. for well structured and detailed discussion of knowledge distillation frameworks [Gou+21].

2.2 Latent Space and Visualization

In statistics, a *latent variable* of a statical model is a hidden variable that is meaningful but not directly observable. It can only be inferred through other observed variables, and one has no direct control over it. In the context of neural nets, the *latent space* or *embedding space* is a lower-dimensional representation of high-dimensional data.

Hinton et al. describe a conceptual block in that “we tend to identify the knowledge in a trained model with the learned parameter values, and this makes it hard to see how we can change the form of the model but keep the same knowledge” [HVD15]. They proposed a more abstract view of knowledge in neural nets where the knowledge instead lies in the learned mapping from input vectors to output vectors. This view entangles the knowledge from the particular parameter instantiation and justifies the knowledge distillation framework.

For a particular deep neural network, different representations of the input can be considered. Every inner layer of an artificial neural network is viewed as a function mapping from one feature space to another. In a well-trained network, the layers must encode an internal representation of the observed data that is meaningful to solve the given task, where the deeper layers are said to operate on more abstract interpretations of the input data. The layers from the first up to the target layer form a mathematical model that does feature extraction from the raw data.

2.2.1 Examples of Latent Spaces

Most of the time we cannot interpret the extracted features directly but items that resemble each other more closely are positioned closer to one another in the latent space. For perception models, there exist methods to visualize the patterns that kernels of CNN layers fire on [ZF14; Wei+15; MV15; OMS17]. These visualizations show that in the first layer convolutional networks detect edges and color gradients, later layers fire on simple geometrical shapes and the final layers work on high-level concepts specific to the dataset e.g., wheels, faces and legs.

A prominent example of latent space in neural networks arises in the generative adversarial network (GAN) architecture where a non-linear mapping from a latent distribution to the real data is learned through adversarial training. From uniformly distributed random noise as input, a generator network produces an output that the discriminator network has to detect as an artificial example. Both networks are trained simultaneously in an adversarial process where the generator has to fool the discriminator. In the unique solution, the generator perfectly recovers the training data distribution and the discriminator guesses with a probability of $\frac{1}{2}$ [Goo+14]. GANs are capable of producing photo-realistic images from randomly sampled latent inputs [BDS18]. A branch of GAN research has been preoccupied with the question of how semantics are organized in the latent space. Radford et al. first showed “interesting vector arithmetic properties” emerging in the generators latent space that enable semantic operations that are not possible in pixel space [RMC16]. By averaging generated images that shared a concept (gender, facial expression, glasses) they were able to produce directions in the latent space that allowed vector arithmetic for visual concepts e.g., subtracting the direction of “neutral woman” from images of smiling women and adding the vector direction of “neutral man” resulted in images of smiling men.

Simple feature space arithmetic was first discovered for word embeddings in the context of natural language [Mik+13]. By predicting the context of words with a skip-gram model in a large corpus *word2vec*’s learn high-quality word embeddings. Using word embeddings rather than one-hot coded vectors helps natural language processing tasks like automatic speech recognition and machine translation achieve better performance by introducing a notion of similarity between words and show interesting characteristics in the latent space.

Adding the vector representing the words “Germany” and “capital” meaningfully combines the concepts and the resulting vector is close to the vector for “Berlin”. Simple arithmetic operations like vec(“King”) - vec(“Man”) + vec(“Woman”) result in a vector whose nearest neighbor is the vector for “Queen” and subtracting the vector representing a country from the vector representing its capital results in a “is capital of” vector that added to different capitals results in a vector close to the embedding of the associated country [MYZ13].

2.2.2 Visualization

Data reduction plays an integral role in both pre-processing datasets and visualizing the results of machine learning models. While high dimensionality can pose a problem to machine learning in terms of computational scalability and the amount of necessary training data, also known as the curse of dimensionality, this thesis uses dimensionality reduction for its capabilities in data visualization. Visualization i.e., reducing the high dimensional data to two or three dimensions that are comprehendible for a human, is an integral part of machine learning not only for building and debugging models but also for understanding the underlying topology of the data.

UMAP (Uniform Manifold Approximation and Projection) is a novel technique for dimension reduction that has a strong theoretical foundation based on manifold theory and topological data analysis [McI+18]. It is assumed that the data approximately lies on a locally connected manifold and is uniformly distributed on that manifold and preserves the topological structure of said manifold putting more focus on the local distance than long-range distances. Both UMAP and the previous state-of-the-art of dimensionality reduction for visualization t-SNE [MH08] are loss functions that make similar points attract each other and push dissimilar points away from each other minimized through gradient descent. Other than t-SNE UMAP is scalable to massive data and able to cope with the diversity of data available from high dimensional raw data input to complex structured feature spaces.

2.3 Dataset and Heterogeneity

2.3.1 The CIFAR Datasets

For his MSc thesis, Alex Krizhevsky created the CIFAR (Canadian Institute for Advanced Research) image datasets from an unlabeled image dataset for unsupervised learning [Kri09]. From millions of unlabeled tiny (32 by 32 pixel) color images 6000 examples (5000 in the training and 1000 in the test set) for each of the 10 classes (airplane, automobile, bird, cat, deer, dog, frog, horse, ship and truck) were combined in the CIFAR-10 dataset (50000 training and 10000 test images). The images in the CIFAR-100 dataset are similar to the CIFAR-10 dataset but per class 600 images (500 training set and 100 testing set) were selected and the 100 classes divided into 10 sub-categories do not overlap with the original classes of CIFAR-10.

2.3.2 Synthetic Heterogeneity

For both, the teacher’s and the student’s training set the CIFAR-10 dataset, which is a popular choice among knowledge distillation as well as federated learning researchers,

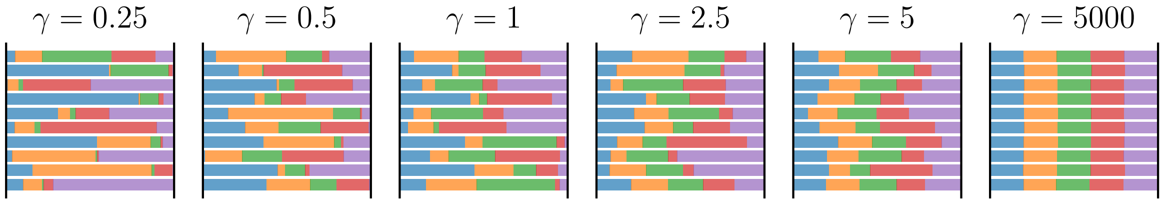


Figure 2.1: Example class populations drawn from a Dirichlet distribution with varying concentration parameter $\gamma \in 0.25, 0.5, 1, 2.5, 5, 5000$

λ	0.25	0.5	1	2.5	5	5000
KL div	0.7733	0.5076	0.3017	0.1369	0.0714	7.448e-05

Table 2.1: Average KL divergence between the drawn populations and the uniform distribution

is used. Because the CIFAR datasets are uniformly distributed over the classes and the subject of this work is to research knowledge distillation under heterogeneous data the datasets are synthetically made non-uniform to model realistic federated learning scenarios. This thesis follows the work of Hsu et al. and generates a non-uniform class distribution with the Dirichlet distribution, a multivariate generalization of the Beta distribution [HQB19]. In the initial federated learning paper, McMahan et al. generated non-iid data by first sorting by class labels then dividing the data into shards and distributing 2 shards to each client. In the end, most clients had only examples of 2 out of 10 classes which is a very extreme scenario. Some datasets have an inherent non-uniformness like the EMNIST [Coh+17] dataset that contains handwritten letters partitioned by the writers.

Since no such natural partition exists for the CIFAR-10 dataset the training sample is constructed in a way that it follows a categorical distribution $\mathbf{q}$ ($\sum_{i=1}^{10} \mathbf{q}_i = 1$) over 10 classes (for clarity this class distribution will be called population) which is drawn from a Dirichlet distribution $\mathbf{q} \sim \mathrm{Dir}(\gamma \mathbf{p})$. The Dirichlet distribution is parameterized by the prior over the classes $\mathbf{p}$ and the concentration parameter γ. Smaller concentration results in populations that are more skewed with the extreme case of $\gamma \to 0$ putting all weight only a single class. Conversely higher concentration results in populations that are more uniform and $\gamma \to \infty$ builds the uniform distribution.

To avoid the case where a population puts no weight on some classes and the student is unaware of those classes every class gets a weight of at least 0.01 for CIFAR-10. Figure 2.1 shows 10 example class populations drawn from a Dirichlet distribution with a uniform prior and concentration $\gamma \in \{0.25, 0.5, 1, 2.5, 5, 5000\}$ (where a concentration of 5000 is similar to the iid case). Non-iid training sets are built by first drawing a population via the Dirichlet method and then picking instances according to the class population constraint. For every training set, two test sets are created one with a uniform and one with a class distribution similar to the training set.

To quantify the similarity of populations drawn with different concentration values Table 2.1 reports the Kullback-Leibler divergence between the populations drawn from the Dirichlet distribution and a uniform distribution averaged over 10000 draws.

Chapter 3

Related Work

3.1 Knowledge Distillation

Knowledge distillation is a more recent addition to the training procedure of neural nets in the sub-fields of model compression and transfer learning. Being concerned with the resource demands of ensemble methods i.e., methods that improve performance by combining results of multiple models, Buciluă et al. train a compact artificial neural net to imitate the function learned by an ensemble of decision trees. They bring the classification quality of ensembles to portable devices or sensor networks, and allow for applications in which real-time predictions are needed [BCN06]. With the focus of the machine learning community switching to neural nets, Ba et al. demonstrate that the knowledge acquired by a large ensemble of neural models can also be transferred to a single small neural net [BC14]. Taking up the idea of model compression pioneered by Rich Caruana and his collaborators, Hinton et al. formulate their concept of knowledge in neural networks and its transfer through distillation from teacher to student. By using soft targets as the knowledge source they show that student improvements through distillation are possible even when the transfer set lacks any examples of one or more of the classes [HVD15]. Searching for architectural conditions that yield optimal compression via distillation Cho et al. find somewhat counter-intuitively that bigger more capable teachers do not necessarily produce better students. They found evidence that early-stopped teachers i.e., teachers that have not reached their full potential, make better teachers and suggest that distillation cannot succeed when student capacity is too low to successfully imitate the teacher [CH19]. A recent publication on fidelity vs generalization in knowledge distillation attends to fact that a “surprisingly large discrepancy between the predictive distributions of the teacher and the student” remains [Sta+21]. It is plausible that conveying latent space knowledge of the teacher to the student during the distillation process could result in higher fidelity by more closely matching the teacher’s abstractions.

Several contributions work on improving the distillation of the teacher model by transferring information about the inner structure of the teacher to the student [Rom+15; KZ17; PT18; Heo+19]. FitNets were the first to deviate from the original response-based knowledge distillation formulation. The authors transfer intermediate representations by introducing hints from the hidden teacher layers that supervise the student’s hidden layers during training. This technique allows them to successfully train thin very deep neural networks [Rom+15]. Komodakis et al. propose attention as a mechanism of transferring knowledge by extracting spatial attention map from the teacher model and distilling the information of where the network focuses to the student [KZ17]. Passalis et al. introduce a probabilistic method for knowledge transfer that matches the probability distribution of the data in the feature space representation rather than just matching the actual latent representation. This allows for cross-modal knowledge transfer and transferring the knowledge of hand-crafted features [PT18]. Heo et al. utilize adversarial examples obtained through an adversarial attack and give the student more accurate information about the decision boundary of the teacher [Heo+19]. The CRD (contrastive representation distillation) framework forges a connection between knowledge distillation and self-supervised representation learning. A contrastive loss that pulls "positive" pairs close and pushes apart the representation between "negative" pairs is said to better capture correlations or higher-order dependencies in representational space [TKI19]. In a similar work at the same intersection of fields, Chen et al. utilize a locality preserving loss that is aimed especially at guiding between layers of different sizes for which techniques like FitNets would need additional fully-connected layers that increase the cost of training [Che+21]. Sarfraz et al. compare many of these different distilling methods and extensively analyze how the underlying mechanisms affect the generalization performance under noisy labels, imbalanced classes and adversarial example transferability [SAZ20].

Another line of research departs from the original compression idea and expands the applications of knowledge distillation [CGS16; Zha+18a; Ani+18]. Net2Net focuses on transferring the knowledge from a previous network to several networks each deeper or wider network than the teacher to speed up the design process and training of larger models [CGS16]. Deep mutual learning lifts the strict distinction between teacher and student and trains an ensemble of students to learn collaboratively. Without a pre-trained teacher, the students teach each other throughout the training process [Zha+18a]. In a similar fashion engineers at Google employ an online variant of distillation, training two networks on disjoint subsets and additionally making them agree on stale predictions of each other. By utilizing parallelization and distribution across machines they were able to fit very large datasets cost-effectively [Ani+18]. This paper and deep mutual learning mark a transition towards a peer collaborative learning scenario like federated learning and are often referred back to by work in the federated distillation community.

3.2 Federated Machine Learning

Distributed machine learning was a precursor to federated machine learning. Dean et al. increase the scale of deep learning by utilizing computing clusters with thousands of machines to run asynchronous stochastic gradient descent for training deep networks one order of magnitude larger than the previous state-of-the-art. Similarly, Anil et al. distribute the training of two large models with asynchronous stochastic gradient descent and additionally use mutual knowledge distillation to keep both models aligned [Ani+18].

Federated machine learning is a new scenario for machine learning in which multiple clients with local data jointly train a deep learning model. Caution is exercised concerning the privacy of each client as the collective trains on their combined data, without any of the participants having to disclose their private data to each other or a centralized authority.

In their influential paper McMahan et al. introduce the federated learning environment and emphasize the benefit of training on real-world data from mobile devices and the privacy sensitivity of such data. Their proposed solution to this predicament is leaving the training data distributed on mobile devices and collaboratively training a global model by sharing model parameters. The global model is held at a central server and iteratively updated by each client on their local data. The client's model parameters are sent to the server with aggregates them — hence the name FedAvg – and publishes the next version of the global model. They mention that by combining the computational resources of the clients' computation becomes essentially free compared to communication costs [McM+17]. While FedAvg works reasonably well, federated learning has to deal with several challenges. Because the quality of the collaboratively learned model is determined by the combined available data of all clients, the collective tends grow possibly to millions of participants [Sat+20b]. The training data collected by the individual clients is influenced by their local environment and usage pattern and both size and distribution will vary across different clients [Sat+20b]. Mobile devices are limited their capacity to run computations and their participation in the collective can generally not be guaranteed due to connectivity and energy reasons. To test the efficiency of federated averaging under non-iid data Zhao et al. artificially partition the data randomly assign each client 2 partitions from 2 classes. They mitigate the performance loss that FedAvg experiences in their experiment by sharing a small subset of data globally between the edge devices [Zha+18b]. Disclosing a subset of private data to have a uniform base dataset seems contrary to the federated learning's commitment to privacy. The method used to introduce heterogeneity into the clients private data is simplistic and not representative for real world scenarios.

Hsu et al. propose a method based on the Dirichlet distribution to synthesize datasets with a continuous range of identicalness which has found recognition in the federated learning community. They evaluated the efficacy of FedAvg under the created heterogeneity condition and found the momentum term of SGD to help ease the negative effects of non-iid data [HQB19]. Sattler et al. show that existing extensions that focus on reducing upstream and downstream communication are very sensitive to non-iid data distributions and propose a new efficient communication protocol for federated learning that resolves these issues [Sat+20b]. Their sparse ternary compression build on top-k sparsification, quantization, optimal lossless coding of the weight updates and a caching mechanism to keep client synchronized converges faster than federated averaging in term of epochs and communicated bits even under heterogeneous data.

Very recently the knowledge distillation mechanism has been introduced to federated machine learning [LW19; Seo+20]. Preoccupied by the fact that a heterogeneous nature of tasks and intellectual property concerns does not permit sharing the model architecture with other clients Li et al. allow clients to have uniquely designed models. The authors leverage knowledge distillation to understand the knowledge of others without sharing data or model architecture and perform federated learning despite each participant having a different model architecture. After training on their private data the clients send soft targets on a shared public dataset to a server that updates the global model to reflect a consensus on the public dataset. The clients then train their local model to approach the consensus global model [LW19]. Delving for a method to reduce the costs of communication in federated learning Sattler et al. show that by compressing the communicated knowledge via quantization and delta-coding a reduction in communication of four orders of magnitude compared to federated averaging can be achieved [Sat+20a].

Chapter 4

Latent Space Knowledge Distillation

This chapter addresses the above-stated problems of federated learning with heterogeneity and scarcity of training data in a knowledge distillation setting with reduced complexity. Focusing on the distillation process while keeping the problem tractable, the federated learning environment is simplified to a single stationary teacher single student knowledge distillation process. In this sense, knowledge distillation is a special case of “federated” learning, where the teacher model resembles the global model and the client trains a student model suited to the particular distribution of private training data.

4.1 Problem Definition

This thesis addresses the intricate effects of heterogeneous data on knowledge distillation and shows that distillation in the latent space can mitigate these problematic effects. Inspired by the complex federated learning scenario, a knowledge distillation scenario where the training data — and thus the task — of teacher and student differ is investigated. While the individual data distribution of the student differs, the tasks are understood as classification tasks in the same domain $\mathbb{D} \ni (\mathbf{x}, \mathbf{y})$, with $\mathbf{x}$ being the input and $\mathbf{y}$ the corresponding class label out of c total classes. The teacher is trained on a *public dataset* $\mathcal{P} \subset \mathbb{D}$ with a uniform class distribution defining the general task. The student holds a *private dataset* $\mathcal{S} \subset \mathbb{D}$ with a non-uniform class distribution that describes its personalized task. Tn the experiments of Chapter 5, to capture the nature of real-world problems where federated learning is applied, the student's datasets are subject to synthetically created heterogeneity and scarcity. The actual distillation process — whether in the latent space or the output space — happens on the *transfer dataset* $\mathcal{O}$. This dataset could be the private dataset $\mathcal{S}$, a different subset of the domain $\mathbb{D}$, or an auxiliary dataset. As the transfer dataset does not need to be labeled, the auxiliary dataset can be comprised of different classes from a related domain.

The objective of the student is to achieve the best classification performance on the private task. By computing the accuracy on a hold out test set with the same class distribution as the private training data $\mathcal{T}_{priv}$, an indication for the performance of the model in practice is measured. It could be that the class distribution at inference changes over time towards the more general task i.e., that represented by the public data. If this is the case, the student wants to choose the distillation method that best distills the teacher on the uniform task and retains this information while adjusting to the private task. Indicating the student's performance on the original task, its accuracy on a uniform

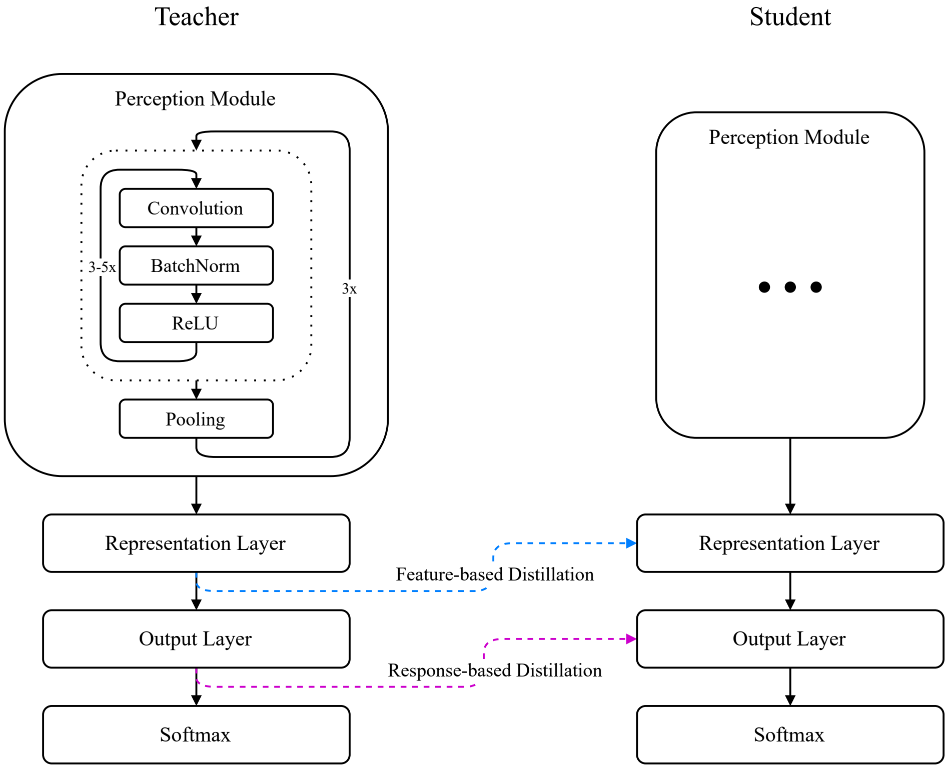


Figure 4.1: Visual comparison of response-based and feature-based distillation for an image classification model. The size indicates that the teacher's perception module is larger but both representation layers have the same size.

test set $\mathcal{T}_{uni}$ is also calculated. In the experiments both accuracies are reported.

4.2 Distillation in the Latent Space

The motivational example highlights the necessity for an effective knowledge distillation process when dealing with non-iid data in the distillation transfer set (Table 1.1). Feature-based distillation extends knowledge distillation by using the output of one or more of the teacher's intermediate layers as the knowledge source to guide the student's training process. Figure 4.1 gives a visual comparison of both feature-based distillation and the vanilla response-based method. Distillation in the embedding space is more in line with the attribution that knowledge of neural nets lies in the learned mappings, the multiple levels of feature representation with increasing abstraction.

The initial representation distillation experiment (Table 1.2) shows that introducing latent space supervision can help when faced with non-iid data. The authors of FitNets [Rom+15] found representation learning to be important for distilling thin very deep neural networks as response-based knowledge fails to address the intermediate-level supervision from the teacher model [Gou+21].

To explain why distillation in the latent space could alleviate the difficulties of knowledge distillation for federated learning, the major data constraints, found in such settings, must be addressed. One part of the scenario at hand is the limited amount of training data. Introducing a distillation loss inside the network i.e., from an inner layer of the teacher to an inner layer of the student, can improve the effectiveness of the training. The loss does not have to flow backward through all the layers, and a larger more expressive gradient reaches the earlier layers as the problem of vanishing gradients is reduced. Standard cross-entropy minimizing training suffers when under the effect of imbalanced class distribution. It exhibits bias towards the prevalent classes at the expense of the minority [SAZ20]. By regularizing the student to match the internal representation of the non-biased teacher, it is to be expected that some of the bias in the student's latent space can be mitigated. At the same time, the student retains the capacity to adjust to the private task distribution because the feature-based distillation loss does not affect the output layer.

4.2.1 Designing the Latent Space

Image classification, the machine learning problem under which this thesis studies the effect of latent space distillation, is the standard task for working with knowledge distillation and one of the most intensively studied machine learning problems in general. The model design can rely on well-studied and known architectural components like the convolution neural networks that have prevailed as the basic building blocks for perception networks. Multiple convolution layers are combined with a non-linear activation function and a down-sampling pooling layer in hierarchically connected blocks. They form the perception module that extract features out of the raw image data. The general structure of the perception architecture is visualized in Figure 4.1.

To distill latent space knowledge from the teacher to the student with a conventional loss function e.g., the mean squared error loss, the embeddings of both models have to have the same size. For response-based distillations, this is given because both models project the input image to the same c output classes. But as the teacher is generally more capable and model variability is a centerpiece of knowledge distillation, a compatible size cannot be assumed. For the latent spaces to have the same dimensionality, a fully connected layer is appended after the convolutional part of the networks. This *representation layer* has the same size l for both the teacher and the student and is the subject of investigation in Section 5.3.

With the same feature size ensured, the mean squared error distillation loss between the representation output of the teacher $r^T(\mathbf{x})$ and the student $r^S(\mathbf{x})$ can be calculated.

$$\mathcal{L}_{distill} = \mathcal{L}_{MSE}(r^S(\mathbf{x}), r^T(\mathbf{x})) = \left|r^S(\mathbf{x}) - r^T(\mathbf{x})\right|^2 \tag{4.1}$$

The distillation loss function $\mathcal{L}_{dist}$ regularizes the student to follow the teacher's representation and thus the internal structure of the teacher. When back-propagated, the loss only effects the parameters up to and including the representation layer. The mean squared error loss brings the intermediate output i.e., after the representation layer, of the student close to the intermediate output of the teacher.

A fully connected layer is usually followed by a non-linear activation function such as the rectified linear unit [NH10] (ReLU) to create non-linear classification boundaries. From a mathematical point of view, two consecutive fully-connected layers without an activation function in between are equivalent to one fully-connected layer i.e., one single linear transformation. In this method, the added representation layer is not followed by an activation function. The additional layer introduced for distillation does not change the optimization topology of the neural net and empirically results in better performance of the distilled student. As such, the proposed technique for creating latent space distillation endpoints has the advantage of not being intrusive by distorting the mathematical topology of the models.

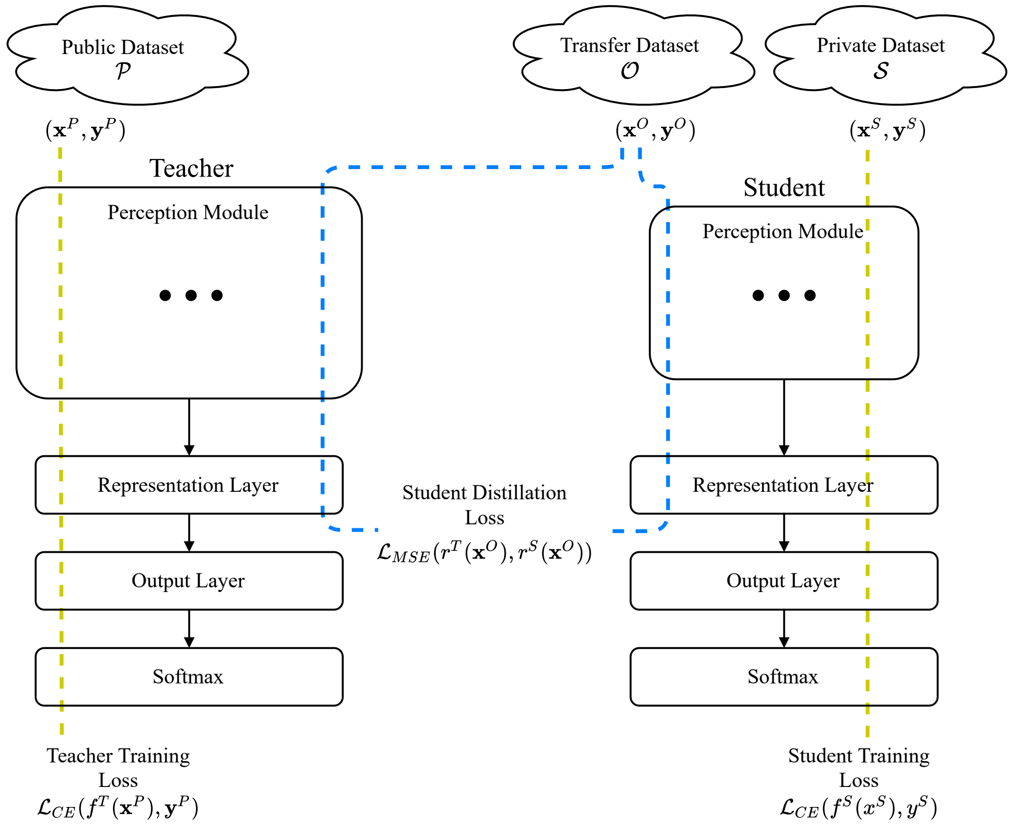

Public Dataset $\mathcal{P}$
$(\mathbf{x}^P, \mathbf{y}^P)$
Transfer Dataset $\mathcal{O}$
$(\mathbf{x}^O, \mathbf{y}^O)$
Private Dataset $\mathcal{S}$
$(\mathbf{x}^S, \mathbf{y}^S)$
Teacher
Perception Module
Representation Layer
Output Layer
Softmax
Student
Perception Module
Representation Layer
Output Layer
Softmax
Student Distillation Loss
$\mathcal{L}_{MSE}(r^T(\mathbf{x}^O), r^S(\mathbf{x}^O))$
Teacher Training Loss
$\mathcal{L}_{CE}(f^T(\mathbf{x}^P), \mathbf{y}^P)$
Student Training Loss
$\mathcal{L}_{CE}(f^S(x^S), y^S)$

Figure 4.2: A schema for Latent Space Distillation vizualising the losses for the teacher and the student.

4.3 Scheme for Heterogeneous Distillation

This section describes the setup for the latent space distillation training pipeline accounting for heterogeneous data of the student. The training pipeline consists of a teacher training phase and a student training phase. The teacher is trained on the public dataset $\mathcal{P}$. The student distills the teacher model on the transfer set $\mathcal{O}$ and is trained on the private dataset $\mathcal{S}$ according to the private task. The code is free-open source software and is available online[1].

First, the teacher model f^S is trained to minimize the cross-entropy loss

$$\text{minimize} \quad \mathcal{L}^T = \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{P}} \mathcal{L}_{CE}(f^T(\mathbf{x}), \mathbf{y}) = -\sum_{(\mathbf{x},\mathbf{y})\in\mathcal{P}} \mathbf{y} \log(f^T(\mathbf{x})) \tag{4.2}$$

with regard to the target label $\mathbf{y}$ on the public data $\mathcal{P}$ for a certain amount of epochs. After the initial training, the teacher model is available to the student, but its parameters are never updated.

As visualized in Figure 4.2, the student is trained with two losses. The losses are calculated on two possibly different datasets, so every training epoch for the student consists of two sub-phases. In an alignment phase, the converged teacher model is used to obtain the representation, which the teacher assigns to an image $r^T(\mathbf{x})$ of the transfer dataset $\mathbf{x} \in \mathcal{O}$. With the latent space output of the teacher model, the student is regularized

[1] https://github.com/matzebond/master-fed

via the distillation loss function to mimic it with its representation $r^S(\mathbf{x})$. The mean squared error distance between both embedding for every image in the transfer dataset is minimized (Equation 4.3). In this phase, the student's latent space is "aligned" with the teacher's latent space.

$$\begin{aligned}\text{minimize} \quad \mathcal{L}^S &= \overbrace{\alpha \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{O}} \mathcal{L}_{MSE}(r^S(\mathbf{x}), r^T(\mathbf{x}))}^{\text{Teacher Alignment}} &&+ \overbrace{\sum_{(\mathbf{x},\mathbf{y})\in\mathcal{S}} \mathcal{L}_{CE}(f^S(\mathbf{x}), y)}^{\text{Private Task}} \\ &= \alpha \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{O}} \left|r^S(\mathbf{x}) - r^T(\mathbf{x})\right|^2 &&- \sum_{(\mathbf{x},\mathbf{y})\in\mathcal{S}} \mathbf{y}\log(f^S(\mathbf{x})) \end{aligned} \tag{4.3}$$

In the private phase, the student is trained on its non-iid private training data $\mathbf{S}$ with a regular cross-entropy loss. The distinction between the alignment and private phase is made to allow for a transfer dataset that is different from the private data.

4.3.1 Weight Freezing

The implemented framework has an option for freezing the parameters of the representation layer and the parameters of the perception module during the student's private phase. If this option is enabled, cross-entropy gradient calculation w.r.t those weights is skipped. They are not updated via stochastic gradient descent in the private training phase. The feature extracting part of the student model is then exclusively adjusted by distilling the representations of the teacher. This option has no effect on the gradients for the parameters in the output layer. They are updated as usual when the output layer is adjusted while the representation is fixed. This option was used in the initial experiment Table 1.2 to retain the pre-distilled representation when training the student.

If the transfer set is the private training set $\mathcal{O} = \mathcal{S}$ and no parameter freezing is desired, the second sub-phase can be merged with the alignment phase. In that case, the distillation loss and the cross-entropy loss are summed during a single iteration over the private dataset.

4.4 Latent Space Augmentation

In an attempt to make latent space distillation more robust, an augmentation technique in the latent space is proposed. In computer vision, augmentation is commonly applied to boost the performance of models by adding rotated, cropped, or colors tinted versions of the original training images to the dataset. With this technique, vision models that are robust against changes in lighting, viewing angle, or other recording conditions missing in the dataset are trained. The technique is related to denoising autoencoders where a model learns to reproduce the original image from a version with added noise. Models which receive noisy inputs have clearer hidden layer features than autoencoders without noise.

These considerations led to the proposed latent space augmentation technique. Instead of the teacher's output $r^T(\mathbf{x})$ for the original image $\mathbf{x} \in \mathcal{O}$, a related representation is given to the student. With a small probability g, the student is given the teacher's intermediate output of an alternative image $\mathbf{x}' \in \mathcal{O}$. This image is from the same class i.e., shows the same object, and its embedding by the teacher $r^T(\mathbf{x}')$ should bear some resemblance to the embedding of the original image. By learning that the original image could also have a slightly different embedding, the student is expected build more robust abstractions that help classification.

Chapter 5

Experimental Evaluation

5.1 Experimental Setup

Most important for this work is the distinction between public data and private data. The teacher's training dataset $\mathcal{P}$ consists of all the data in the CIFAR10 dataset. Its class distribution is uniform over all classes and contains no bias. In federated learning, diverse clients hold heterogeneous data cooperate, and the student's task is related to the task of the teacher model. The image-label pairs in the student's private data are sampled heterogeneously from CIFAR10 to simulate more accurate federated learning conditions. The class distribution q is drawn from a Dirichlet Distribution $\mathbf{q} \sim \mathrm{Dir}(\gamma \mathbf{p})$ with the concentration parameter γ controlling the skewed-ness the distribution and $\mathbf{p} = [1, 1, \dots, 1] \in \mathbb{R}^c$ representing the uniform class prior. According to the class distribution q, m training instances are sampled from the CIFAR10 dataset. The private set is also used as the transfer set $\mathcal{O} = \mathcal{S}$. Note that in a real-world scenario, the private data would most likely not overlap with the public data. Because the particular class distribution has a huge effect on the nature and the difficulty of the task, all experiments are repeated five times with different class distributions, and the mean is reported.

The key feature of knowledge distillation is that it allows the training of models with different computational complexity for devices with varying compute power. Latent space distillation is not dependent on a specific architecture. The technique works with various networks as it only slightly alters the network topology. The authors of FitNets proposed four performance-efficient architectures, which recently were also used by Chen et al. in their locality preserving distillation experiments. In this thesis, because the teacher model was hard to replicate, the most complex student model is used as the teacher. Studying the effect of different student-teacher capacity gaps is beyond the scope of this thesis, and the least complex model is used as the only student. The consequences of capacity mismatches are not as relevant because, with scarce data, the student is not expected to train to its full potential [CH19]. The architecture contains an additional representation layer as introduced in Section 4.2.1 and batch normalization layers [IS15] after every convolutional layer is used. Batch normalization has found empirical success in neural networks by fixing the distribution of each layer's inputs during training to reduce the internal covariate shift. Figure 4.1 shows the structure of the model and the perception module in detail, and 5.1 reports the exact sizes of the individual layers. The default size for the representation layer l is 500 neurons. The teacher has roughly 10 times the parameters of the student network and uses 12.6 times as many multiply-accumulate floating-point operations per inference.

teacher	student
conv 3x3x32	conv 3x3x16
conv 3x3x32	conv 3x3x16
conv 3x3x32	conv 3x3x16
conv 3x3x48	max-pool 2x2
conv 3x3x48	
max-pool 2x2	
conv 3x3x80	conv 3x3x32
conv 3x3x80	conv 3x3x32
conv 3x3x80	conv 3x3x32
conv 3x3x80	max-pool 2x2
conv 3x3x80	
conv 3x3x80	
max-pool 2x2	
conv 3x3x128	conv 3x3x48
conv 3x3x128	conv 3x3x48
conv 3x3x128	conv 3x3x64
conv 3x3x128	max-pool 8x8
conv 3x3x128	
conv 3x3x128	
max-pool 8x8	
fc → l	fc → l
fc → 10	fc → 10
softmax	softmax

Table 5.1: Design parameters of teacher and student architectures using l as the size for the representation layer and the softmax function as defined in Equation 2.1 before the output.

For comparison, all students use the same teacher model for distillation. The model was trained for 200 epochs on the full CIFAR-10 dataset with the Adam optimizer [KB17], an initial learning rate of 0.0001, and a batch size of 32. All images undergo per channel normalization, and during training the dataset is augmented by random horizontal flipping and center cropping to increase the performance of the teacher model.

The students train for 300 epochs aligning through latent space distillation and matching the original labels on the private set $\mathcal{S}$. The distillation loss function for both latent space distillation (Equation 4.3) and vanilla distillation (Equation 4.1) is the mean squared error function. In both loss summation, the distillation loss is weighted by $\alpha = 2$ which empirically gave the best results. The temperature parameter τ for vanilla distillation is set to 5. Similar to the training of the teacher model, the batch size is set to 32, and the Adam optimizer with an initial learning rate of 0.0001 is used. The weight freezing option described in Section 4.3.1 is not enabled by default.

5.2 Data Scarcity and Heterogeneity

This section experimentally explores the specific conditions under which students with non-iid data can be trained effectively with the proposed latent knowledge distillation method. The total amount and heterogeneity of the private data are varied, simulating different levels of data scarcity and task divergence. The heterogeneity of the class distribution $\mathbf{q}$ is adjusted by the concentration parameter of the Dirichlet distribution ($\gamma \in \{0.25, 0.5, 1, 2.5, 5\}$). Different amounts of private data ($m \in \{512, 1024, 2048, 4096\}$)

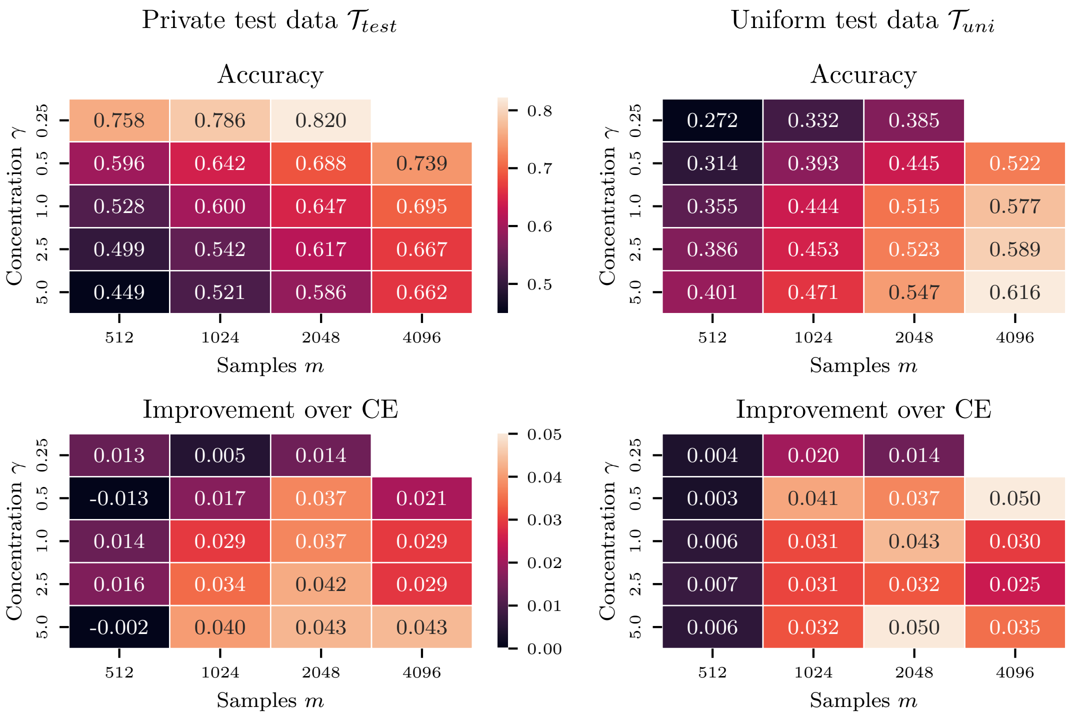


Figure 5.1: Performance of latent space knowledge distillation effected by data scarcity and data heterogeneity. Mean accuracy (top) and improvement in accuracy over a CE baseline (bottom) on private test data (left) and uniform test data (right) averaged over 5 runs.

are sampled from CIFAR10 according to the class distribution.

The results of this experiment are presented in a heatmap in Figure 1. The x-axis displays the amount of private data, the y-axis the heterogeneity of the student's data, and the color intensity expresses the performance. The second row of heatmaps visualizes the improvements over the regular training with cross-entropy on the private data $\mathcal{S}$. As expected, the accuracy on both test sets correlates positively with the size of the training set. The more training data available, the better the distillation of the teacher and the adjustment to the private task. More imbalance in the training data leads to a more biased, and thus easier to solve, problem on the non-iid test data $\mathcal{T}_{priv}$. Learning from massively heterogeneous data makes the student one-sided and it performs worse when presented with the uniform test set $\mathcal{T}_{uni}$. Contrarily, the students with more regular training data perform better on the uniform test data that somewhat matches their private data and worse on the non-iid test data that represent a more difficult because less biased task. Comparing latent space knowledge distillation with regular training, the students with more than 512 data points in the training set achieve a significant improvement over the baseline. The improvement margin gets smaller with increasing data heterogeneity, suggesting that distillation of a good representation gets more difficult when the transfer set $\mathcal{O} = \mathcal{S}$ is different from the training set of the teacher model $\mathcal{P}$.

The previous experiment shows that distilling the teacher with heterogeneous data is especially difficult when the dataset for distillation is small. To improve students that do not have enough private data for successful distillation, a second publicly available dataset is used as the transfer set $\mathcal{O}$. Internal structure of the teacher is distilled with

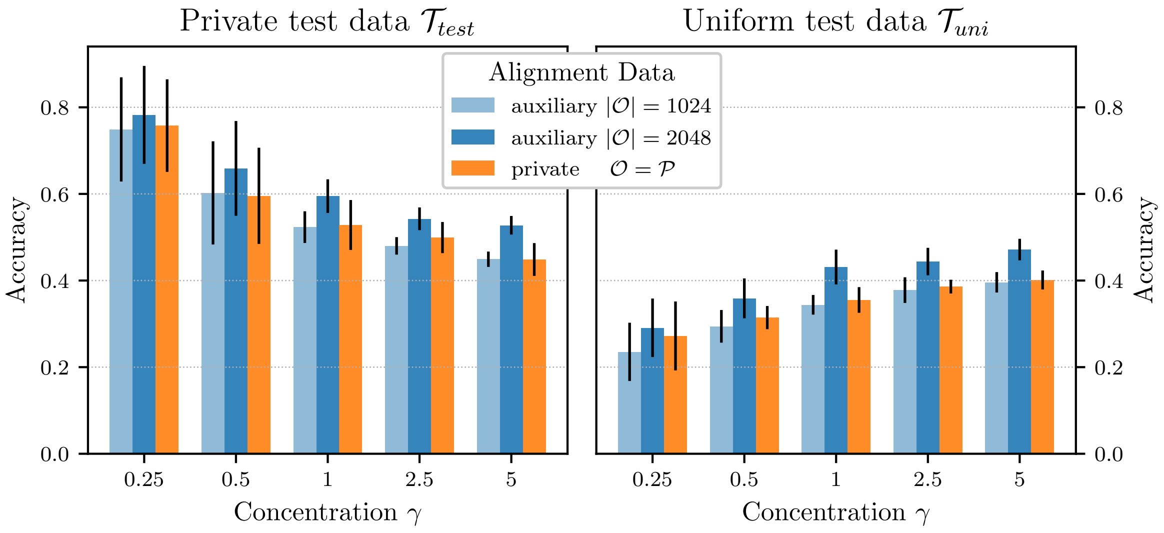


Figure 5.2: Comparing effect of alignment data source and size for student with scarce private data (512 data points). Public data is uniformly sampled from CIFAR-100. One σ error bars over 5 runs.

this auxiliary dataset, and the private data is used to adjust to the personalized task. For this experiment, the CIFAR100 dataset constitutes the publicly available auxiliary data. The image classes in the CIFAR100 are distinct from the classes in the training data of the teacher and student. For response-based knowledge distillation, the images in the transfer set would somewhat questionably be mapped to the different output classes of the training set. When distilling in the latent space, the classification output is ignored, and instead, the intermediate representation is the distillation target. Evaluating how big the transfer set needs to be to improve distillation, only a subset of the CIFAR100 dataset is used. Figure 5.2 reports the results for students with $m = 512$ private images and use 2048 (1024) images CIFAR100 dataset for distillation in the alignment phase. More data in the transfer allows more accurate teacher distillation and consequently results in higher accuracy. Since the auxiliary images differ from the original dataset, they are not optimal for distilling a representation train under a different image distribution. Distillation on a transfer set with 1024 images is not enough to improve over distillation solely on the smaller but more accurate private dataset. The abundance of auxiliary data proves to be an advantage when using a larger transfer set. The students with 2048 CIFAR100 images outperform the students distilling from the private data consistently overall levels of data heterogeneity. The experiment shows that in cases of extreme data scarcity distillation on a related dataset improves the accuracy of the student. With an auxiliary dataset to better distill the representation of the teacher, latent space knowledge distillation is possible in scenarios of severe data scarcity.

5.3 Latent Space Dimensionality

This section concentrates on different configurations of the latent space and the impact on the proposed distillation algorithm. To focus on the structure of the latent space, the federated scenario is fixed. All experiments use the same data heterogeneity with a

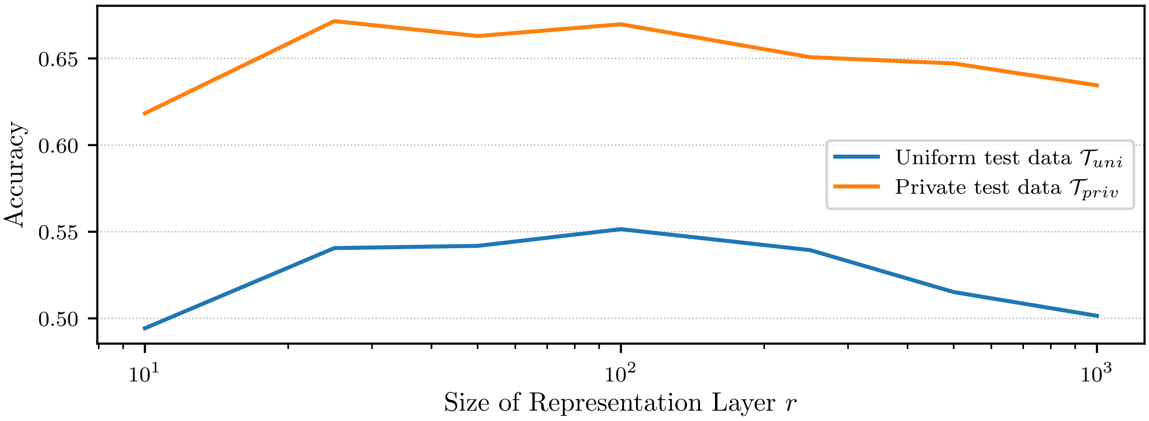


Figure 5.3: Student accuracy after distilling in latent space of different dimensionality $l \in \{10, 20, 50, 100, 250, 1000\}$.

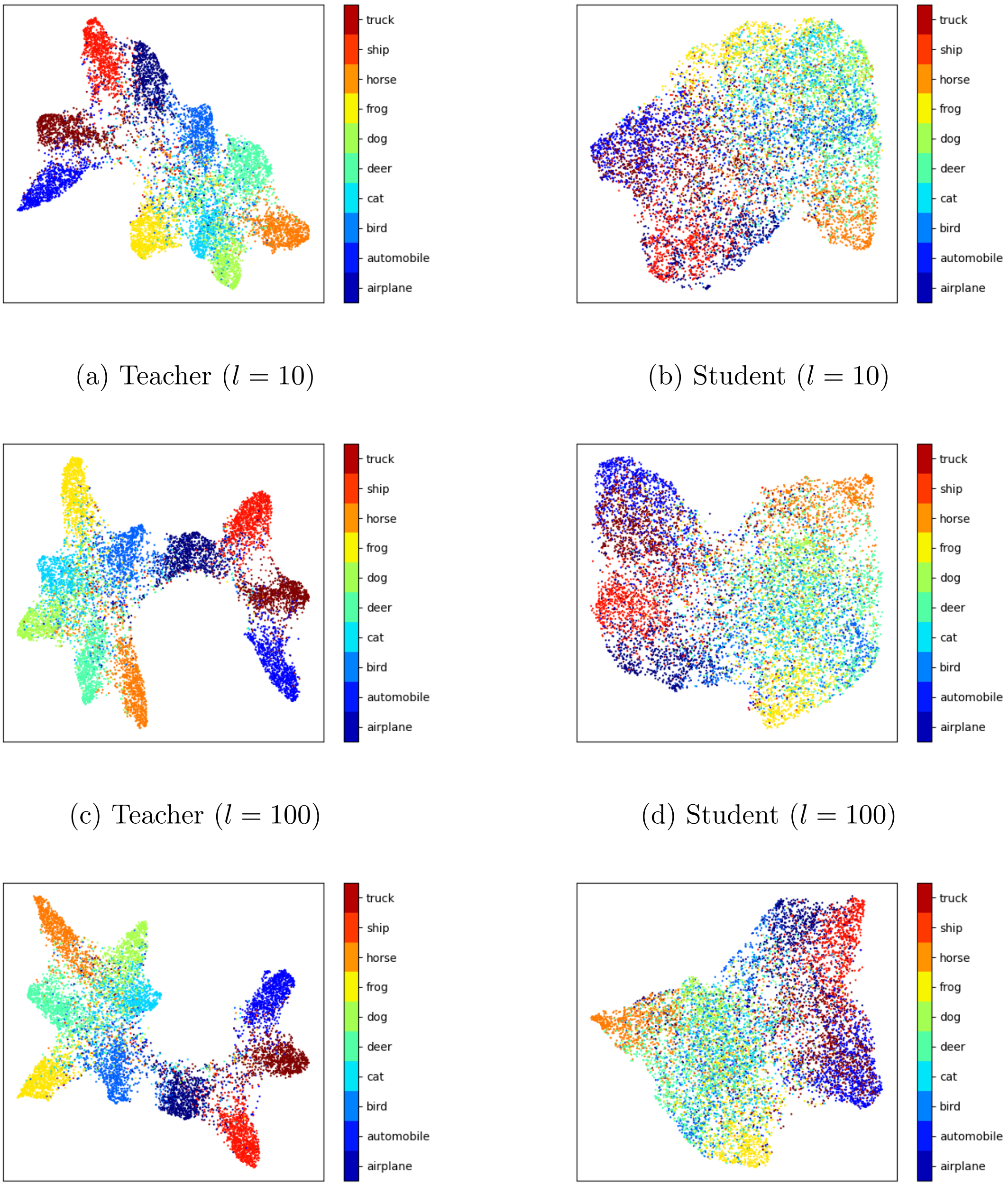
Dirichlet distribution concentration of 1.0 and scarcity with a private training set size of 2048.

To analyze the influence of the dimensionality of the latent space, in the first experiment of this section, the performance of latent space distillation with representation layers of different sizes is evaluated. Figure 5.3 plots the accuracy of distillation for representation layers with output size in the range of 10 to 1000. It shows that for too small and too large layer size, the performance on both test sets degrades. The size of the representation layer does not influence the performance of the teacher model because the representation layer does not include a non-linear activation function. Together with the output layer constitutes a single linear function. With the configuration reported in Table 5.1, the representation layer of the teacher model receives a 128-dimensional input. In the scenario at hand, distillation works best in the latent space of dimensionality 100. The negative effect of higher dimensional features for distillation suggests that the additional capacity for richer abstractions does not compensate for the harder distillation in high dimensional spaces. A linear transformation into a lower dimension loses too much expressiveness and does not benefit the process. The experiment shows that distillation works best in a latent space with a dimensionality similar to the output of the perception module.

The visualizations of the features generated by three teacher models and three student models with different latent space dimensionality are illustrated in Figure 5.4. As the size of the representation layer does not influence the teacher's performance, the projected embeddings of all teachers look similar; aside from the fact that the UMAP method does not fix the orientation. The student in the middle row with a representation layer size of 100 outperforms the others by 4-5%. The features generated by this student have less overlap between the classes and show a cleaner distinction between machinery and the animal images.

5.4 Feature Augmentation

The following experiment evaluates the augmentation technique proposed in Section 4.4. Figure 5.5 shows the performance of student models that distill the teacher in the latent space with a varying amount of latent space augmentation. A higher probability of



(a) Teacher ($l = 10$)

(b) Student ($l = 10$)

(c) Teacher ($l = 100$)

(d) Student ($l = 100$)

(e) Teacher ($l = 1000$)

(f) Student ($l = 1000$)

Figure 5.4: UMAP projection of the features of teachers and students with different representation layer size l.

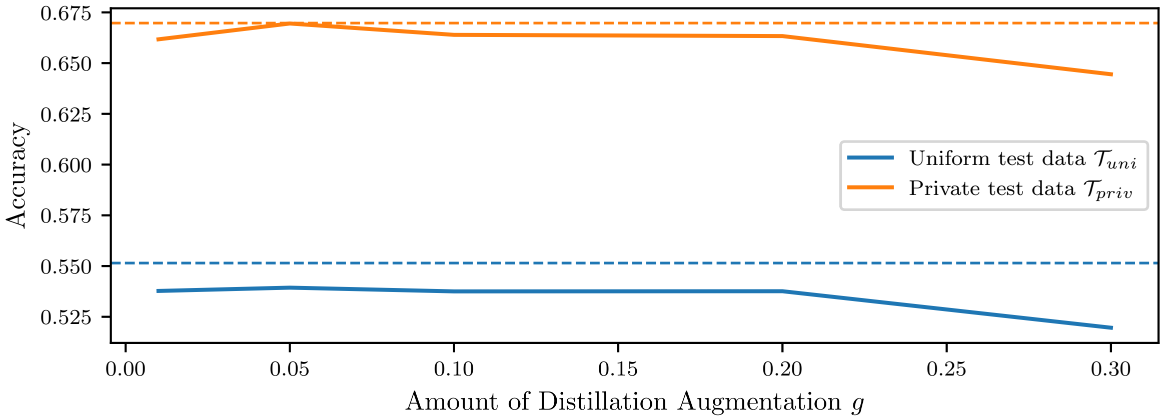


Figure 5.5: Accuracy of latent space distillation with varying amount of latent space augmentation $g \in \{0.01, 0.05, 0.1, 0.2, 0.3\}$. The dashed lines represent the base case without distillation.

augmentation g correlates with worse performance on both the private and the uniform test set. The proposed variant of augmentation for distillation does not build more robust models. The student accuracy is impaired when the teacher's latent space outputs are replaced with similar representations during training.

5.5 Comparison

The last experimental section compares latent space knowledge distillation with other variants of knowledge distillation under the influence of heterogeneous data. Apart from the proposed latent space knowledge distillation(LKD) and the vanilla knowledge distillation(KD), as formulated by Hinton et al., two other feature-based variants of knowledge distillation are compared.

Contrastive representation distillation (CRD) [TKI19] proposed by Tian et al. utilizes a contrastive loss that pulls "positive" pairs close and pushes apart the representation between "negative" pairs. Positive pairs are the teacher's representation and the student's representation of the same image. Negative pairs are the representations of both models on different inputs. After the training, the embedding of the student should be close to the corresponding embedding of the teacher At the same time, the student's embedding should be far apart from the teacher's embeddings of other images. This technique from self-supervised learning is said to capture correlations or higher-order dependencies in the latent space adequately. The last variant for comparison is Locality Preserving Distillation (LP) [Che+21]. From manifold learning, the authors take the idea of locally representing the manifold by reconstructing each input point as a weighted combination of its neighbors. With a loss that pulls the student's embeddings of images together, depending on the local relationship between the teacher's embeddings, the local structure of the features is retained. Because it is impractical to analyze all embeddings for a single embedding, only the relationship with the k-nearest neighbors in the current mini-batch in the teacher's latent space are considered. Both CRD and LP have evolved from a different context than latent space knowledge distillation. The difference is that they do not require the feature size of teacher and student to be the same. Their main goal is to preserve the structure

Test data	LKD	KD	CRD	LP
Private $\mathcal{T}_{priv}$	**0.674**	0.631	0.655	0.391
Uniform $\mathcal{T}_{uni}$	0.544	**0.551**	0.530	0.183

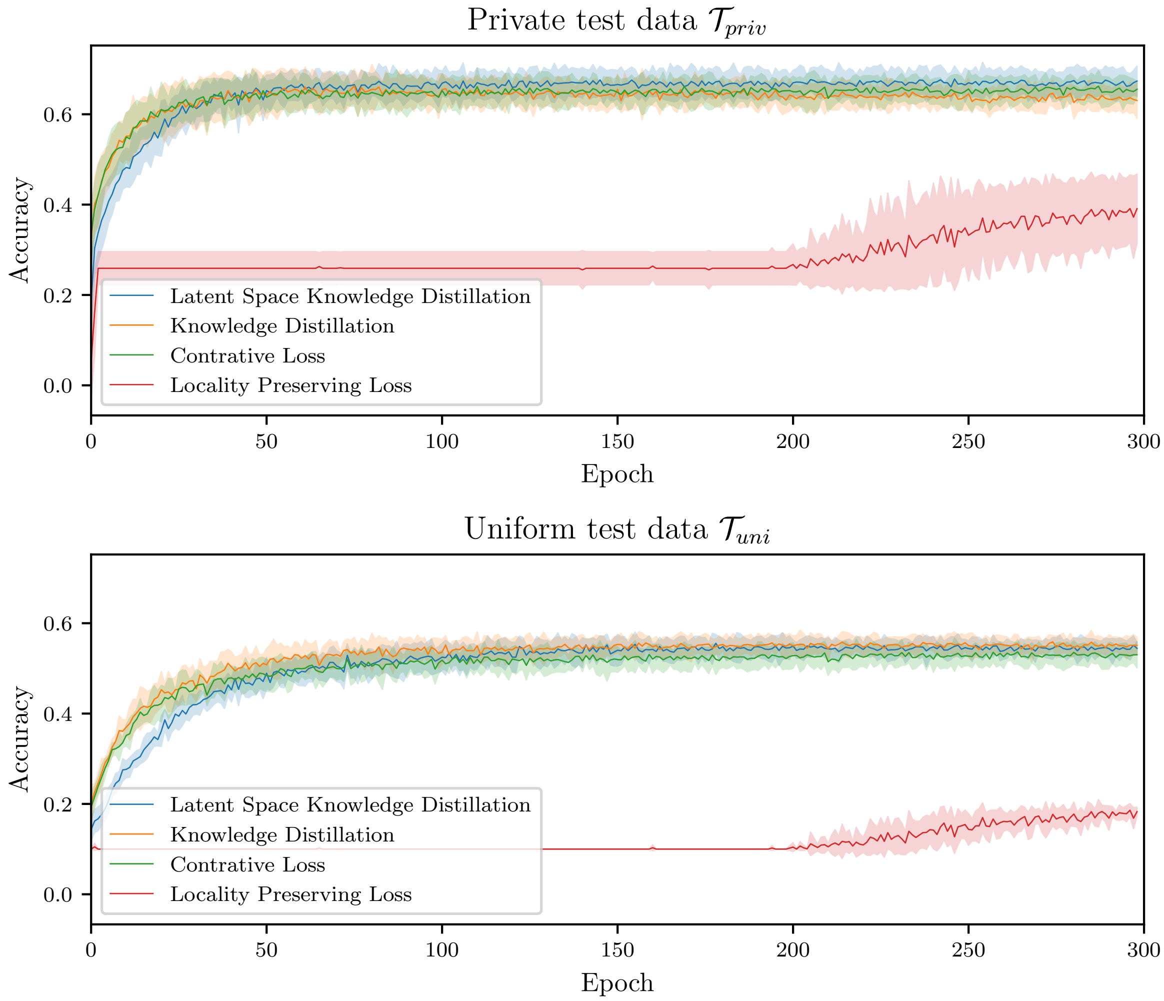
Table 5.2: Accuracy of different distillation methods on private and uniform test data

of the features in the teacher's latent space in the student's embeddings of usually lower dimensionality.

Table 5.2 presents the results of the four different methods under a non-iid scenario with private data of size 2048, Dirichlet heterogeneity of 1.0, and a latent space dimensionality of 100 in both teacher and student. CDR is evaluated with the contrastive temperature of 1 and LP with the five nearest neighbors. For the same experiment, Figure 5.6 shows the convergence of the different methods over the 300 training epochs of the students. It shows clearly that LP suffers from a class collapse under heterogeneous data. After the second epoch, only the predominant class of the distribution is predicted and the method reaches an accuracy of exactly 0.1 on the uniform data. After epoch 200 LP slowly recovers from this collapse by increasing the locality preserving loss in favor of decreasing the cross-entropy loss but still has the worst performance after 300 epochs. On the private test set in the end both LKD and CRD outperform vanilla knowledge distillation but LKD takes around 50 epochs to reach surpass KD. The contrastive variant does not perform as well on the uniform test data where LKD and KD reach roughly a 2% advantage over CRD. In the given scenario latent space knowledge is the best of the compared methods.

5.6 Summary

The experiments show that latent space distillation can handle scenarios with heterogeneous data better than vanilla knowledge distillation or regular training. Other methods that distill the teacher's representations do not perform better when the teacher's and the student's features are the same size. In scenarios where the student has enough private data, the proposed distillation method works well by distilling the teacher on the private dataset. When the private data is too scarce, using auxiliary data to distill the teacher improves latent space distillation. For the best performance, the added representation layer should roughly match the output size of the teachers perception model. The proposed latent space augmentation during distillation does not increase the performance of the resulting student models.



Private test data $\mathcal{T}_{priv}$

Uniform test data $\mathcal{T}_{uni}$

Figure 5.6: Convergence of different distillation methods in a scenario with 2048 private data-points and a heterogeneity concentration of 1.0

Chapter 6

Conclusion

In this thesis a novel distillation technique that mitigates against the problems of distillation in federated machine learning scenarios with scarce and heterogeneous data is developed. Knowledge distillation has been successfully applied in federated contexts to protect the confidentiality of the participant's data. But as shown in two initial experiments, the standard version of knowledge distillation does not produce adequate student models when confronted with challenging, highly heterogeneous environments. All experiments were conducted in a student-teacher scenario with a synthetically introduced class imbalance to capture the essence of heterogeneity in federated learning.

The proposed latent space distillation technique is resilient against the challenges of diverse heterogeneous settings. By appending a fully-connected layer to the feature extracting part of both student and teacher, the feature size of both models coincides. The student's latent space is brought into alignment with the latent space of the teacher through a mean squared error distillation loss function. The introduced technique compares favorably to other distillation variants, improving the accuracy on the individual task of the student while retaining proficiency on the original assignment of the teacher.

Through extensive experimental evaluation, latent space distillation was shown to operate effectively under different levels of data scarcity and data heterogeneity. When the student's data was too scarce to support the proper distillation of the teacher, data from an auxiliary dataset supported the distillation process. The developed technique proved effective in versatile heterogeneity scenarios and is a suitable addition to the knowledge distillation toolbox.

6.1 Outlook

The proposed technique facilitates the utilization of knowledge distillation under heterogeneous data constraints in a single student single teacher context. The transfer to full-scale federated scenarios remains to be investigated. With more participants collaborating, more diverse models will be encountered. Under these conditions, distillations variants that do not require the same feature size for all models might be at an advantage. Combining techniques from representation learning like the contrastive loss with federated knowledge distillation could foster further advances in the field.

In a digression from the main subject, this work also explored the possibility of augmentation in the latent space. While replacing the distillation target with a related embedding was not beneficial, more sophisticated ideas like interpolation or denoising in the latent space could produce more robust models. Investigating additional forms of representation augmentation proposes an intriguing direction for further research.

On a more philosophical level, cooperation and the ability to share knowledge between individual agents have been detrimental to the advancement of intelligent life. While machines are increasingly outperforming humans on isolated tasks, truly knowledgeable machines will need a source of collective decentralized intelligence to fulfill their potential. Knowledge distillation is a first attempt to make artificial agents capable of sharing knowledge and learning from each other and could prove to be a crucial concept for the further development and interconnection of intelligent machines.

List of Tables

1.1 Experimental results showing the difficulties of knowledge distillation with data heterogeneity 5

1.2 Representation distillation on the full dataset 7

2.1 Average KL divergence between the drawn populations and the uniform distribution 13

5.1 Design parameters of teacher and student architectures using l as the size for the representation layer and the softmax function as defined in Equation 2.1 before the output. 26

5.2 Accuracy of different distillation methods on private and uniform test data 32

List of Figures

1.1 Fundamental concept of Latent Space Distillation 6

2.1 Example class populations drawn from a Dirichlet distribution with varying concentration parameter $\gamma \in 0.25, 0.5, 1, 2.5, 5, 5000$ 13

4.1 Visual comparison of response-based and feature-based distillation for an image classification model. The size indicates that the teacher's perception module is larger but both representation layers have the same size. 20

4.2 A schema for Latent Space Distillation vizualising the losses for the teacher and the student. 22

5.1 Performance of latent space knowledge distillation effected by data scarcity and data heterogeneity. Mean accuracy (top) and improvement in accuracy over a CE baseline (bottom) on private test data (left) and uniform test data (right) averaged over 5 runs. 27

5.2 Comparing effect of alignment data source and size for student with scarce private data (512 data points). Public data is uniformly sampled from CIFAR-100. One σ error bars over 5 runs. 28

5.3 Student accuracy after distilling in latent space of different dimensionality $l \in \{10, 20, 50, 100, 250, 1000\}$. 29

5.4 UMAP projection of the features of teachers and students with different representation layer size l. 30

5.5 Accuracy of latent space distillation with varying amount of latent space augmentation $g \in \{0.01, 0.05, 0.1, 0.2, 0.3\}$. The dashed lines represent the base case without distillation. 31

5.6 Convergence of different distillation methods in a scenario with 2048 private data-points and a heterogeneity concentration of 1.0 33

6.1 Convergence of the first motivational experiment. Standard cross-entropy (CE) and vanilla knowledge distillation (KD) evaluated on private test data (upper) and test data that is uniformly distributed (lower) with one σ error bands. 45

6.2 Convergence of the second motivational experiment. Accuracy of training after distilling teacher's output vs teacher's representation (one σ error bands). 46

Bibliography

[Ani+18] Rohan Anil et al. “Large Scale Distributed Neural Network Training through Online Distillation”. In: International Conference on Learning Representations. Feb. 15, 2018 (cit. on p. 16).

[BC14] Jimmy Ba and Rich Caruana. “Do Deep Nets Really Need to Be Deep?” In: NIPS. Jan. 1, 2014 (cit. on pp. 9, 15).

[BCN06] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. “Model Compression”. In: *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD ’06. New York, NY, USA: Association for Computing Machinery, Aug. 20, 2006, pp. 535–541. DOI: 10.1145/1150402.1150464 (cit. on pp. 9, 15).

[BDS18] Andrew Brock, Jeff Donahue, and Karen Simonyan. *Large Scale GAN Training for High Fidelity Natural Image Synthesis*. Version 1. Sept. 28, 2018. arXiv: 1809.11096 [cs, stat] (cit. on p. 11).

[CGS16] Tianqi Chen, Ian Goodfellow, and Jonathon Shlens. *Net2Net: Accelerating Learning via Knowledge Transfer*. Apr. 23, 2016. arXiv: 1511.05641 [cs] (cit. on p. 16).

[CH19] Jang Hyun Cho and Bharath Hariharan. “On the Efficacy of Knowledge Distillation”. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019, pp. 4794–4802 (cit. on pp. 15, 25).

[Cha+16] William Chan et al. “Listen, Attend and Spell: A Neural Network for Large Vocabulary Conversational Speech Recognition”. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Mar. 2016, pp. 4960–4964. DOI: 10.1109/ICASSP.2016.7472621 (cit. on p. 3).

[Che+21] Hanting Chen et al. “Learning Student Networks via Feature Embedding”. In: *IEEE Transactions on Neural Networks and Learning Systems* 32.1 (Jan. 2021), pp. 25–35. DOI: 10.1109/TNNLS.2020.2970494 (cit. on pp. 16, 25, 31).

[Coh+17] Gregory Cohen et al. *EMNIST: An Extension of MNIST to Handwritten Letters*. Mar. 1, 2017. arXiv: 1702.05373 [cs] (cit. on p. 13).

[Dea+12] Jeffrey Dean et al. “Large Scale Distributed Deep Networks”. In: *NIPS*. 2012 (cit. on p. 16).

[Goo+14] Ian Goodfellow et al. “Generative Adversarial Nets”. In: *Advances in Neural Information Processing Systems* 27 (2014), pp. 2672–2680 (cit. on p. 11).

[Gou+21] Jianping Gou et al. “Knowledge Distillation: A Survey”. In: *International Journal of Computer Vision* (Mar. 22, 2021). DOI: 10.1007/s11263-021-01453-z (cit. on pp. 9, 10, 20).

[Heo+19] Byeongho Heo et al. “Knowledge Distillation with Adversarial Samples Supporting Decision Boundary”. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 33.01 (01 July 17, 2019), pp. 3771–3778. DOI: 10.1609/aaai.v33i01.33013771 (cit. on pp. 15, 16).

[How+17] Andrew G. Howard et al. *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*. Apr. 16, 2017. arXiv: 1704.04861 [cs] (cit. on p. 3).

[HQB19] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. *Measuring the Effects of Non-Identical Data Distribution for Federated Visual Classification*. Sept. 13, 2019. arXiv: 1909.06335 [cs, stat] (cit. on pp. 13, 17).

[Hub+17] Itay Hubara et al. “Quantized Neural Networks: Training Neural Networks with Low Precision Weights and Activations”. In: *The Journal of Machine Learning Research* 18.1 (Jan. 1, 2017), pp. 6869–6898 (cit. on p. 3).

[HVD15] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. “Distilling the Knowledge in a Neural Network”. In: *NIPS Deep Learning and Representation Learning Workshop*. 2015 (cit. on pp. 9, 11, 15, 31).

[IS15] Sergey Ioffe and Christian Szegedy. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. Mar. 2, 2015. arXiv: 1502.03167 [cs] (cit. on p. 25).

[Ita+21] Sohei Itahara et al. *Distillation-Based Semi-Supervised Federated Learning for Communication-Efficient Collaborative Training with Non-IID Private Data*. Jan. 20, 2021. arXiv: 2008.06180 [cs] (cit. on p. 4).

[Jum+21] John Jumper et al. “Highly Accurate Protein Structure Prediction with AlphaFold”. In: *Nature* 596.7873 (2021), pp. 583–589. DOI: 10.1038/s41586-021-03819-2. pmid: 34265844 (cit. on p. 3).

[KB17] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. Jan. 29, 2017. arXiv: 1412.6980 [cs] (cit. on p. 26).

[Kim+21] Taehyeon Kim et al. *Comparing Kullback-Leibler Divergence and Mean Squared Error Loss in Knowledge Distillation*. May 19, 2021. arXiv: 2105.08919 [cs] (cit. on p. 10).

[Kri09] Alex Krizhevsky. *Learning Multiple Layers of Features from Tiny Images*. 2009 (cit. on pp. 5, 12).

[KZ17] Nikos Komodakis and Sergey Zagoruyko. “Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer”. In: *ICLR*. Paris, France, June 2017 (cit. on pp. 15, 16).

[LW19] Daliang Li and Junpu Wang. *FedMD: Heterogenous Federated Learning via Model Distillation*. Oct. 8, 2019. arXiv: 1910.03581 [cs, stat] (cit. on p. 17).

[McI+18] Leland McInnes et al. “UMAP: Uniform Manifold Approximation and Projection”. In: *Journal of Open Source Software* 3.29 (Sept. 2, 2018), p. 861. DOI: 10.21105/joss.00861 (cit. on p. 12).

[McM+17] Brendan McMahan et al. “Communication-Efficient Learning of Deep Networks from Decentralized Data”. In: *Artificial Intelligence and Statistics*. Artificial Intelligence and Statistics. PMLR, Apr. 10, 2017, pp. 1273–1282 (cit. on pp. 13, 16, 17).

[MH08] Laurens van der Maaten and Geoffrey Hinton. “Visualizing Data Using T-SNE”. In: *Journal of Machine Learning Research* 9.86 (2008), pp. 2579–2605 (cit. on p. 12).

[Mik+13] Tomas Mikolov et al. “Distributed Representations of Words and Phrases and Their Compositionality”. In: *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*. NIPS’13. Red Hook, NY, USA: Curran Associates Inc., Dec. 5, 2013, pp. 3111–3119 (cit. on p. 11).

[MV15] Aravindh Mahendran and Andrea Vedaldi. “Understanding Deep Image Representations by Inverting Them”. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015, pp. 5188–5196 (cit. on p. 11).

[MYZ13] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. “Linguistic Regularities in Continuous Space Word Representations”. In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. NAACL-HLT 2013. Atlanta, Georgia: Association for Computational Linguistics, June 2013, pp. 746–751 (cit. on p. 12).

[NH10] Vinod Nair and Geoffrey E. Hinton. “Rectified Linear Units Improve Restricted Boltzmann Machines”. In: *Proceedings of the 27th International Conference on International Conference on Machine Learning*. ICML’10. Madison, WI, USA: Omnipress, June 21, 2010, pp. 807–814 (cit. on p. 21).

[OMS17] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. “Feature Visualization”. In: *Distill* 2.11 (Nov. 7, 2017), e7. DOI: 10.23915/distill.00007 (cit. on p. 11).

[PT18] Nikolaos Passalis and Anastasios Tefas. “Learning Deep Representations with Probabilistic Knowledge Transfer”. In: Proceedings of the European Conference on Computer Vision (ECCV). 2018, pp. 268–284 (cit. on pp. 15, 16).

[RMC16] Alec Radford, Luke Metz, and Soumith Chintala. “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”. In: *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2016 (cit. on p. 11).

[Rom+15] Adriana Romero et al. “FitNets: Hints for Thin Deep Nets”. In: *ICLR 2015*. ICLR 2015. San Diego, California, May 7, 2015. arXiv: 1412.6550 (cit. on pp. 15, 20).

[RSN20] R. K. Ramakrishnan, E. Sari, and V. P. Nia. “Differentiable Mask for Pruning Convolutional and Recurrent Networks”. In: *2020 17th Conference on Computer and Robot Vision (CRV)*. 2020 17th Conference on Computer and Robot Vision (CRV). May 2020, pp. 222–229. DOI: 10.1109/CRV50864.2020.00037 (cit. on p. 3).

[San+18] Mark Sandler et al. "MobileNetV2: Inverted Residuals and Linear Bottlenecks". In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018, pp. 4510–4520 (cit. on p. 3).

[Sat+20a] Felix Sattler et al. *Communication-Efficient Federated Distillation.* Dec. 1, 2020. arXiv: 2012.00632 [cs, stat] (cit. on p. 17).

[Sat+20b] Felix Sattler et al. "Robust and Communication-Efficient Federated Learning From Non-i.i.d. Data". In: *IEEE Transactions on Neural Networks and Learning Systems* 31.9 (Sept. 2020), pp. 3400–3413. DOI: 10.1109/TNNLS.2019.2944481 (cit. on p. 17).

[SAZ20] Fahad Sarfraz, Elahe Arani, and Bahram Zonooz. *Knowledge Distillation Beyond Model Compression.* July 3, 2020. arXiv: 2007.01922 [cs, stat] (cit. on pp. 16, 21).

[Seo+20] Hyowoon Seo et al. *Federated Knowledge Distillation.* Nov. 4, 2020. arXiv: 2011.02367 [cs, math] (cit. on p. 17).

[SMS20] Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. "Clustered Federated Learning: Model-Agnostic Distributed Multitask Optimization Under Privacy Constraints". In: *IEEE Transactions on Neural Networks and Learning Systems* (2020), pp. 1–13. DOI: 10.1109/TNNLS.2020.3015958 (cit. on p. 4).

[Sta+21] Samuel Stanton et al. *Does Knowledge Distillation Really Work?* June 10, 2021. arXiv: 2106.05945 [cs, stat] (cit. on p. 15).

[TKI19] Yonglong Tian, Dilip Krishnan, and Phillip Isola. "Contrastive Representation Distillation". In: International Conference on Learning Representations. Sept. 25, 2019 (cit. on pp. 16, 31).

[Vas+17] Ashish Vaswani et al. "Attention Is All You Need". In: *Advances in Neural Information Processing Systems* 30 (2017), pp. 5998–6008 (cit. on p. 3).

[Wei+15] Donglai Wei et al. "Understanding Intra-Class Knowledge Inside CNN". In: *CoRR* abs/1507.02379 (2015). arXiv: 1507.02379 (cit. on p. 11).

[ZF14] Matthew D. Zeiler and Rob Fergus. "Visualizing and Understanding Convolutional Networks". In: *Computer Vision – ECCV 2014*. Ed. by David Fleet et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2014, pp. 818–833. DOI: 10.1007/978-3-319-10590-1_53 (cit. on p. 11).

[Zha+18a] Ying Zhang et al. "Deep Mutual Learning". In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018, pp. 4320–4328 (cit. on pp. 4, 16).

[Zha+18b] Yue Zhao et al. *Federated Learning with Non-IID Data.* June 2, 2018. arXiv: 1806.00582 [cs, stat] (cit. on p. 17).

[ZXX18] Shiyu Zhou, Shuang Xu, and Bo Xu. *Multilingual End-to-End Speech Recognition with A Single Transformer on Low-Resource Languages.* June 13, 2018. arXiv: 1806.05059 [cs, eess] (cit. on p. 3).

Appendix A

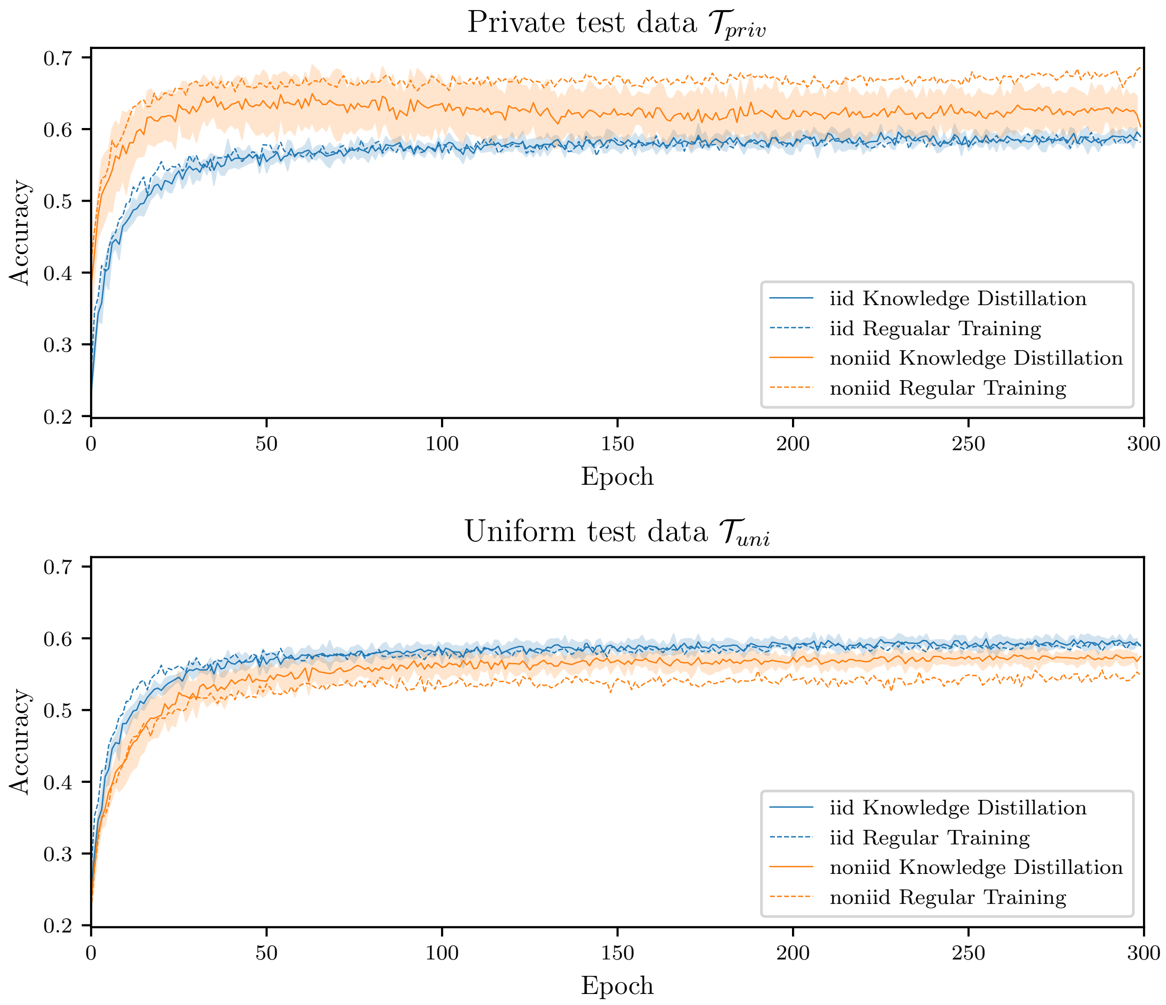


Figure 6.1: Convergence of the first motivational experiment. Standard cross-entropy (CE) and vanilla knowledge distillation (KD) evaluated on private test data (upper) and test data that is uniformly distributed (lower) with one σ error bands.

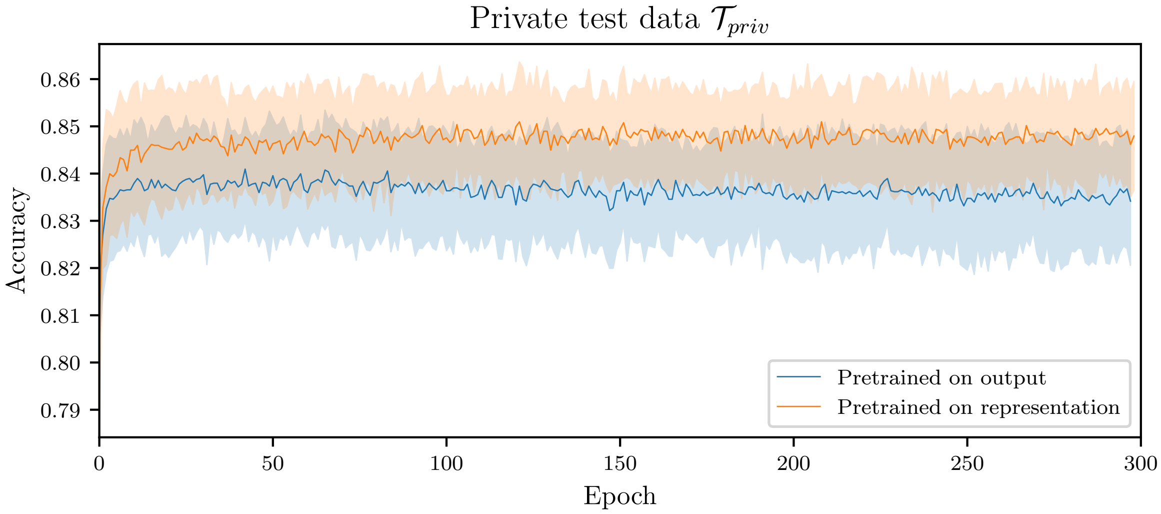


Figure 6.2: Convergence of the second motivational experiment. Accuracy of training after distilling teacher's output vs teacher's representation (one σ error bands).