Technische Universität München
School of Computation, Information and Technology
Department of Informatics

Exposé — Bachelor's Thesis

Adaptation Techniques for using NAS Methods in the FL Setting

Author:	Max Coetzee
Supervisor:	M.Sc. Nick Henze
Advisor:	Dr.-Ing. Niclas Kannengießer
Date:	tbd.

Contents

1 Problem Statement

Engineering the architecture of a neural network for a Deep Learning application is traditionally done by a team experts via a process of trial and error. To reduce the amount of manual labour involved in this process, researchers invented *Neural Architecture Search* (NAS) [2] methods and improved them over the past decade. NAS methods employ diverse strategies to automatically search for a neural network architecture for a given Deep Learning application.

Independantly, but in parallel to NAS, researchers developed a distributed machine learning approach called *Federated Learning* (FL) [10] in response to growing concerns about data privacy. In FL, *clients* collaboratively train a model without sharing their local data. This enhances the privacy of clients' data by ensuring model trainers can not view clients' data and client data is not collected at a central location where a single breach could expose the data of all clients.

Engineering neural network architectures in FL is as labour-intensive as in centralised Deep Learning, therefore researchers have started investigating the use of NAS methods in FL [4] [1] [9], creating *Federated Neural Architecture Search methods* (FedNAS methods) [4].

FedNAS methods also provide a potential alternative to selecting a fixed architecture upfront — a so-called *prededfined architecture*. Predefined architectures can lead to slow training convergence and poorly performing models in FL, because model developers can not view the clients' data and client hardware capabilities vary. Model developers may therefore select a predefined architecture that contains components irrelevant for generalising well from client data sets or select an architecture that trains slowly on some clients. Work has already been done that shows the use of NAS methods can mitigate these issues [5] [11] [12].

The approach taken by most FedNAS methods (and on which we focus in this thesis) is to use NAS methods developed in a centralised setting, which has received significantly more research attention, and use them for FL. However, this requires modifying the NAS method for the FL setting, because many assumptions that hold for the search process in centralised NAS do not hold for FedNAS. These assumption discrepancies result in *challenges* for using centralised NAS methods in FL, and developers have created *adaptation techniques* for overcoming them.

The subset of challenges faced by these kinds of FedNAS methods depends on the specific configuration of FL system parameters [7], such as the degree of heterogeneity in hardware capabilities of clients, average network latency of clients, the number of participating clients etc. The literature identifies two common types FL system parameter configurations, called FL settings: the *cross-device* setting, wherein clients are edge devices, and the *cross-silo* setting, wherein clients are entire organisations, but even within these settings, there is significant variation in the system parameters. Depending on the targeted configuration of FL system parameters, centralised NAS assumptions are violated to a different extent, making some challenges more relevant to them than others. For example, in the cross-silo setting, clients can be expected to be equipped with GPUs, making the challenge described above less relevant.

Developers adapting a centralised NAS method to FL with a specific set of FL system parameters in mind must decide which challenges to prioritise and which adaptation techniques to implement, but the literature does not offer clear advice for these design decisions. The incurred challenges are scattered throughout the literature, the adaptation techniques used to address them are often not presented in isolation and many FedNAS papers do not clearly state the targeted FL system parameters. As a result, developers struggle to assess the usefuleness of existing adaptation techniques for their targeted FL system parameters and risk selecting ineffective techniques or select techniques for addressing a challenge that are known to worsen another.

Prior literature surveys [13] [8] [3] only summarise FedNAS methods on the whole and do not focus on individual, re-usable adaptation techniques. Additionally, prior surveys are limited by the FedNAS methods available at the time or exclude a large share of the FedNAS literature due to their chosen focus. As a result, prior surveys don't provide an exhaustive overview of adaptation techniques and do not help developers assess the usefuleness of existing adaptation techniques for their targeted FL system parameters.

To ease the adaptation of centralised NAS methods to FL in the future, we set out to answer our research question:

What challenges arise from adapting centralised NAS methods to FL and which techniques address these challenges in the literature?

To tackle our research question, we conduct a systematic literature review of papers that present FedNAS methods which modify centralised NAS methods in response to the FL setting.

We define a set of fine-grained parameters to characterise the targeted FL setting of each FedNAS method based on observations of varying setting parameters in the literature. With the help of this characterisation, we identify the violated centralised NAS assumptions and catalogue the challenges that arise from them. Next, we extract unrefined adaptation techniques from the FedNAS methods and iteratively refine and merge them (similar to [6]) to obtain a set of collectively exhaustive adaptation techniques. We analyse how each adaptation technique works towards, against, or does not affect each challenge, and present our findings in the form of a discussion for each adaptation technique, as well as an overview table.

Our review aims to support the creation of new FedNAS methods by developers. By identifying the source of challenges and elaborating on them, we provide clarity on the expected challenges for a targeted FL setting. Based on the expected challenges, FedNAS developers can use our overview of adaptation techniques to guide the design of new FedNAS methods and determine whether to re-use existing techniques, extend them, or develop new ones.

2 Objectives

The primary objective of this thesis is to provide an overview of techniques used to adapt centralised NAS to FL, spawning the following sub-objectives:

1. Identify FL system parameters relevant for adapting a centralised NAS method to FL.
2. Catalogue the assumptions that hold for centralised NAS, but are violated in FedNAS and to what extent these assumptions are violated based on FL system parameters.
3. Describe the challenges arising from the mismatch in assumptions between centralised NAS and FedNAS.
4. Catalogue the adaptation techniques used to overcome these challenges.

3 Explanation of Terms

3.1 Neural Architecture Search (NAS)

Traditionally, neural network architectures are designed by a team of domain and Deep Learning experts. In NAS, the architecture is automatically searched by continuously evaluating the performance of candidate architectures and updating the architectures high performance for a given task, dataset, and constraints. Most NAS methods can be described by their *search space* of possible architectures, the employed *search strategy*, and the *performance estimation strategy*. TODO: supernetwork and hypernetworkbased don't neatly fall into category

3.2 Centralised NAS

3.3 Federated Learning

FL is a machine learning approach in which multiple *clients* collaboratively train a model while keeping training data local to the clients. The training is coordinated by a central *server* that initiates *communication rounds*

3.4 FL system parameters

- number of clients - degree of variance in hardware of clients - average networking latency of clients - average computing power of clients - degree of data imbalance between clients - availability of clients

3.5 Federated Neural Architecture Search (FedNAS)

3.6 Adaptation technique

An adaptation technique is any modification to a NAS method that is explicitly motivated by the FL setting (or by a FedNAS challenge) and is intended to make the search feasible, efficient, or effective in that setting. This thesis conceptualises adaptation techniques at a level where they can be reused as design building blocks across methods.

4 Research Approach

We take a qualitiative research approach in the style of the CDML paper.

4.1 Data Collection

We obtain the set of literature relevant to our review by searching the abstract, title and keywords of literature in Scopus [TODO: cite] with the search string "federated learning neural architecture search" and include literature that presents a FedNAS method that uses a centralised NAS method explicitly modified for FL. We exclude literature that (i) designs FedNAS methods from scratch, (ii) performs only hyperparameter optimisation, (iii) or does does not provide sufficient methodological detail to extract adaptation techniques.

We extend this initial set of literature by recursively adding literature from the references of the literature that meets our inclusion and exclusion criteria until we obtain a set of literature for which no new literature can be added.

4.2 Data Analysis

We follow a qualitative research approach in the form of a systematic literature review. In our review, we develop a conceptual model that links FL system parameters to violated centralised NAS assumptions, the resulting challenges, and the adaptation techniques used to overcome these challenges as follows:

1. **Define Relevant FL System Parameters:** For each FedNAS method, we extract the reported FL system parameters. If parameters are not stated, we infer them from the experimental setup and mark them as inferred.

2. **Challenges arising from Assumption Discrepancies:** We first define challenges that arise for centralised NAS methods in FL, because of assumptions that hold for the centralised NAS, but do not hold for FL.

3. **Unrefined Adaptation Technique Extraction:** We perform open coding on each FedNAS method to extract unrefined adaptation techniques. Any modification to a NAS method that is explicitly motivated by the federated setting is initially coded as one unrefined adaptation technique.

4. **Adaptation Techniques Coneceptualization:** We iteratively refine and merge the unrefined adapatation techniques (similar to [6]) in an axial coding step to obtain a coherent set of adaptation techniques. Unrefined adaptation techniques with conceptually highly-similar mechanisms are merged into a single representative adaptation technique.

5. **Discuss FedNAS Challenges for Adaptation Techniques:** We discuss how each adaptation technique works towards, against, or does not affect overcoming each of FedNAS challenges.

6. **Table Overview:**

   Finally, we create a table that researchers can use to make design decisions about the techniques they wish to use to adapt a NAS method to FL.

   The table contains a coded vector of effects over the FedNAS challenges for each *adaptation technique* based on the prior discussion. new FedNAS methods, they can use this table to choose existing adaptation techniques relevant to the set of FedNAS challenges they need to address for their use case.

1. **Categorise Adaptation Techniques:** After merging, in a second axial coding step, we cluster adaptation techniques based on a) the FedNAS challenges they address and b) the conceptual similarity of their mechanisms. As a result, we obtain a taxonomy of adaptation techniques.

5 Structure

1. **Introduction** *(3 pages)*

   What is the motivation behind this thesis? What is the relevance of this thesis? What problem does this thesis try to solve? What kind of approach does this thesis take to solve that problem? How is the thesis structured?

2. **Background** *(6 pages)*

   What background knowledge is required to understand this thesis?

   2.1 Neural Architecture Search (2.5 pages)

   What is NAS? What is it used for? What is a NAS method? What are the origins of NAS? What environment are NAS methods typically developed in?

   2.2 Federated Learning (2.5 pages)

   What is Federated

   2.3 Federated Neural Architecture Search (0.5 pages)

   How do FedNAS methods relate to NAS and FL? What are the origins FedNAS methods and how have they been developed over the last couple of years?

3. **Method** *(5 pages)*

   How does this thesis achieve the objectives described in Section 2?

   3.1 Method and Literature Selection (3 pages)

   3.2 Reviewed Literature (2 pages)

4. **Challenges with Adapting Centralised NAS methods to FL** (10 pages):

   4.1 Relevant FL system parameters *(3 pages)*

   What makes a FL system parameter relevant to adaptation techniques described in this thesis? Which FL system parameters are relevant and which ones are not?

   4.2 Assumption Discrepancies between NAS and FedNAS *(3 pages)*

   What assumptions can be made in

   4.3 Adaptation Challenges (4 pages) How do FL system parameters influence challenges faced for adapting centralised NAS methods to ?

5. **Adaptation Techniques** (25 pages) What adaptation techniques are described in the literature? What can we learn from these adaptation techniques that we can use for adapting NAS methods to FL in the future? For which FL system parameter configuration was each adaptation technique developed? How do adaptation techniques work towards, against, or do not affect overcoming each of the challenges relevant to that FL system parameter configuration?
   5.1 Adaptation Technique 1 *(1 page)*

   Description of the adaptation technique, the targeted FL system parameter configuration, the FedNAS methods that use it and a discusson on how it overcomes challenges.

   5.2 Adaptation Technique 2 *(1 page)*

   5.3 ...

   5.4 Adaptation Technique 20 *(1 page)*

   5.5 Overview *(5 pages)* A large overview table that makes it easy to see which adaptation techniques are beneficial for overcoming which challenges at a glance.
6. **Discussion** *(2 pages)*
7. **Conclusion** *(1 page)*

Totalling roughly *... pages*

6 Expected Results

- challenges for FedNAS methods based on FL system parameters - catalog of adaptation techniques - bibliography of FedNAS methods

7 Open Issues and Problems

- finding the correct abstraction level for adaptation techniques

References

[1] L. Dudziak, S. Laskaridis, and J. Fernandez-Marques. *FedorAS: Federated architecture search under system heterogeneity*. 2022. arXiv: `2206.11239 [cs.LG]`.

[2] T. Elsken, J. H. Metzen, and F. Hutter. “Neural architecture search: A survey.” In: *Journal of Machine Learning Research* 20.55 (2019), pp. 1–21.

[3] M. Hartmann, G. Danoy, and P. Bouvry. *Multi-objective methods in Federated Learning: A survey and taxonomy*. 2025. arXiv: `2502.03108 [cs.LG]`.

[4] C. He, M. Annavaram, and S. Avestimehr. *Towards Non-I.I.D. and invisible data with FedNAS: Federated deep learning via neural architecture search*. 2021. arXiv: `2004.08546 [cs.LG]`.

[5] M. Hoang and C. Kingsford. “Personalized Neural Architecture Search for Federated Learning.” In: *1st NeurIPS Workshop on New Frontiers in Federated Learning (NFFL 2021)* ().

[6] D. Jin, N. Kannengießer, S. Rank, and A. Sunyaev. “Collaborative distributed machine learning.” In: 57.4 (Dec. 2024). ISSN: 0360-0300. DOI: `10.1145/3704807`.

[7] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, R. G. L. D’Oliveira, H. Eichner, S. E. Rouayheb, D. Evans, J. Gardner, Z. Garrett, A. Gascón, B. Ghazi, P. B. Gibbons, M. Gruteser, Z. Harchaoui, C. He, L. He, Z. Huo, B. Hutchinson, J. Hsu, M. Jaggi, T. Javidi, G. Joshi, M. Khodak, J. Konečný, A. Korolova, F. Koushanfar, S. Koyejo, T. Lepoint, Y. Liu, P. Mittal, M. Mohri, R. Nock, A. Özgür, R. Pagh, M. Raykova, H. Qi, D. Ramage, R. Raskar, D. Song, W. Song, S. U. Stich, Z. Sun, A. T. Suresh, F. Tramèr, P. Vepakomma, J. Wang, L. Xiong, Z. Xu, Q. Yang, F. X. Yu, H. Yu, and S. Zhao. *Advances and Open Problems in Federated Learning*. 2021. arXiv: `1912.04977 [cs.LG]`.

[8] S. Khan, A. Rizwan, A. N. Khan, M. Ali, R. Ahmed, and D. H. Kim. “A multi-perspective revisit to the optimization methods of Neural Architecture Search and Hyper-parameter optimization for non-federated and federated learning environments.” In: *Computers and Electrical Engineering* 110 (2023), p. 108867. ISSN: 0045-7906. DOI: `https://doi.org/10.1016/j.compeleceng.2023.108867`.

[9] J. Liu, J. Yan, H. Xu, Z. Wang, J. Huang, and Y. Xu. “Finch: Enhancing federated learning with hierarchical neural architecture search.” In: *IEEE Transactions on Mobile Computing* 23.5 (2024), pp. 6012–6026. DOI: `10.1109/TMC.2023.3315451`.

[10] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. “Communication-efficient learning of deep networks from decentralized data.” In: *Proceedings of the 20th international conference on artificial intelligence and statistics*. Ed. by S. Aarti and Z. Jerry. Vol. 54. Proceedings of machine learning research. PMLR, Apr. 2017, pp. 1273–1282.

[11] E. Mushtaq, C. He, J. Ding, and S. Avestimehr. *SPIDER: Searching personalized neural architecture for federated learning*. 2021. arXiv: 2112.13939 [cs.LG].

[12] J. Yan, J. Liu, H. Xu, Z. Wang, and C. Qiao. "Peaches: Personalized federated learning with neural architecture search in edge computing." In: *IEEE Transactions on Mobile Computing* 23.11 (2024), pp. 10296–10312. DOI: 10.1109/TMC.2024.3373506.

[13] H. Zhu, H. Zhang, and Y. Jin. "From federated learning to federated neural architecture search: a survey." In: *Complex and Intelligent Systems* 7.2 (Apr. 2021), pp. 639–657. ISSN: 2198-6053. DOI: 10.1007/s40747-020-00247-z.