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 of 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) [3] 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.

Independently, but in parallel to NAS, researchers developed a distributed machine learning approach called *Federated Learning* (FL) [12] 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 that FL model trainers cannot view clients' data and that 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 practitioners[1] have started investigating the use of NAS methods in FL [6] [1] [11], creating *Federated Neural Architecture Search methods* (FedNAS methods) [6]. FedNAS methods also provide a potential alternative to selecting a fixed architecture upfront — a so-called *predefined architecture*. Predefined architectures can lead to slow training convergence and poorly performing models in FL, because practitioners cannot view the clients' data, and client hardware capabilities vary. Practitioners 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 [7] [13] [14].

The approach used by most FedNAS methods — and on which we focus in this thesis — is to take NAS methods developed in a centralised setting 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 practitioners have created *adaptation techniques* for overcoming them. For example, the FedNAS method *FedorAS* [1] makes use of the centralised NAS method *SPOS* [4]. Since centralised NAS can assume worker nodes are connected via low-latency, high-bandwidth links, but clients in FL are not, the following challenge arises: sending the parameters of the entire supernet used in SPOS to each client would take an infeasible amount of time. Instead, *FedorAS* adapts *SPOS* such that only a small, sampled subspace of the supernet is sent to each client for training and evaluation.

[1] In this thesis, *practitioners* referes to both users and developers of FedNAS methods.

The relevant subset of challenges faced by these kinds of FedNAS methods depends on the configuration of FL system parameters [9]. System parameters include the average hardware capabilities of clients, the average network latency of clients, the number of participating clients, etc. For example, in the so-called *cross-silo setting*, clients can be expected to be equipped with GPUs, making challenges caused by low-end hardware in FL less relevant.

Practitioners 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. However, the literature does not offer clear guidance 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 a large portion of the FedNAS literature does not clearly state the targeted FL system parameters. As a result, practitioners struggle to assess the usefulness of existing adaptation techniques for their targeted FL system parameters and risk selecting ineffective techniques or selecting techniques for addressing a challenge that are known to worsen another.

Prior literature surveys [15] [10] [5] only summarise FedNAS methods on the whole and do not focus on isolated, 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 do not provide an exhaustive overview of adaptation techniques and do not help practitioners assess the usefulness of existing adaptation techniques for their targeted FL system parameters. Since prior surveys do not solve the problem mentioned in the previous paragraph, we pose the following research question to help with the adaptation of NAS methods to FL:

What challenges arise when adapting centralised NAS methods to FL? How is the relevance of challenges influenced by FL system parameters, and how do adaptation techniques in the literature overcome these challenges?

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

We collect relevant literature by searching for titles, abstracts and keywords in Scopus [2] using the search string "federated learning neural architecture search". We include FedNAS literature that explicitly modifies a centralised NAS method for FL, and exclude FedNAS methods built from scratch, concern only hyperparameter optimisation, or methods with insufficient detail. We then recursively add eligible literature from references.

We conduct a systematic qualitative literature review, building a conceptual model linking violated centralised NAS assumptions to FL challenges, the effect of FL system parameters on challenge relevance, and adaptation techniques. We extract adaptation techniques through open coding and iteratively refine them afterwards. We map each technique's effect onto each challenge and summarise results in a challenge-technique table for practitioners.

Our review aims to support the adaptation of centralised NAS methods to FL. To this end, we make the following contributions:

1. We help practitioners understand the challenges arising from discrepancies in assumptions between centralised NAS and FedNAS.

2. We help practitioners understand which challenges are relevant for their targeted FL system parameters.

3. By extracting adaptation techniques and highlighting the challenges they address, we enable practitioners to reuse existing adaptation techniques for adapting centralised NAS methods to FL for their specific set of challenges.

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. 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.

2. Describe the challenges arising from the mismatch in assumptions between centralised NAS and FedNAS.

3. 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 for a specific Deep Learning application. In NAS, the architecture is automatically searched by continuously evaluating the performance of candidate architectures and updating the architectures with high performance for a given task.

NAS methods differ in their *search space* for potential candidate architectures, the *search strategy* they employ to find an optimal architecture, and the *performance estimation strategy* used by the search strategy to make decisions on which candidates to favour or explore next.

3.2 Federated Learning

FL is a machine learning approach in which multiple *clients* collaboratively train a shared model while keeping training data local to the clients. A central *server* coordinates *communication rounds* with the clients, wherein each client trains the model for several epochs locally and finally sends gradient updates to the server for aggregation into the shared model.

3.3 Adaptation technique

An adaptation technique is a modification to a centralised NAS method that is explicitly motivated by making the NAS method feasible for FL. We refer to the example from the problem statement: The FedNAS method *FedorAS* [1] makes use of the centralised NAS method *SPOS* [4], but runs into the following challenge caused by the fact that centralised NAS can assume worker nodes are connected via low latency, high-bandwidth links, whereas clients in FL are not: sending the parameters of the entire supernet used in SPOS to each client would take an infeasible amount of time. Instead, *FedorAS* adapts *SPOS* such that only a small, sampled subspace of the supernet is sent to each client for training and evaluation.

4 Research Approach

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 [2] 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, or (iii) 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 employ a qualitative research approach, specifically a systematic literature review. In our review, we develop a conceptual model that links violated centralised NAS assumptions to resulting challenges for adapting a centralised NAS method to FL, and the adaptation techniques used to overcome these challenges as follows:

1. **Challenges arising from Assumption Discrepancies:** We first define challenges that arise from using centralised NAS methods in FL, because of assumptions that hold for centralised NAS, but not for FL.
2. **Influence of FL System Parameters on Challenges:** We then classify the influence of FL system parameters on the relevance of challenges as either "increasing" or "decreasing" the relevance of each challenge.
3. **Adaptation Technique Conceptualisation:** Next, we extract adaptation techniques from the literature in two steps.
   3.1 First, 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.
   3.2 Then, we iteratively refine and merge the unrefined adaptation techniques (similar to [8]) to obtain a coherent set of adaptation techniques. Unrefined adaptation techniques with conceptually highly similar mechanisms are merged into a single representative adaptation technique.
4. **Discuss FedNAS Challenges for Adaptation Techniques:** Afterwards, we discuss how each adaptation technique works towards, against, or does not affect overcoming each of the FedNAS challenges.
5. **Table Overview:** Finally, we create a table that contains a coded vector of effects over the challenges for each adaptation technique based on the prior discussion.

Practitioners can use our classification from 2. to first identify challenges relevant to their targeted FL system parameters and then refer to our table from 5. to gain an overview at a glance of which adaptation techniques could be useful to them.

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 used for, and how does it work? 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 Learning? What was it designed for and how does it work?

   2.3 Federated Neural Architecture Search (0.5 pages)

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

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

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

   3.1 Reviewed Literature *(2 pages)* Will include visualisations that break down the included literature for the reader.

   Which literature is included in the review?

   3.2 Methodology *(3 pages)*

   How is the literature analysed? Which methodology is employed and how?

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

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

   How do centralised NAS and FedNAS differ? How does this difference manifest in different assumptions?

   4.2 Adaptation Challenges *(4 pages)*

   How do FL system parameters influence challenges faced in adapting centralised NAS methods to FL?

4.3 Influence of FL system parameters on Challenges *(3 pages)*

Which FL system parameters influence the degree to which a challenge is relevant? Which FL system parameters are relevant and which ones are not?

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 refined adaptation technique, the FedNAS methods that use it and a discussion 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)*

   What do our findings indicate for the future of adapting centralised NAS methods to FL? How can researchers and practitioners use these contributions when adapting centralised NAS methods to targeted FL system parameters?

7. **Conclusion** *(1 page)*

   What is the answer to the thesis's research question, and what are the main contributions?

52 pages in total.

6 Expected Results

1. A catalogue of challenges that arise when adapting centralised NAS methods to FL.

2. Classification of the influence of FL system parameters on the relevance of a challenge.
3. A catalogue of adaptation techniques extracted from FedNAS methods with an in-depth discussion of the challenges they overcome.
4. A tabular overview of adaptation techniques that developers can use to see which adaptation techniques can be used to overcome which challenges to what extent at a glance.

References

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

[2] Elsevier. *Scopus*. Abstract and citation database. Accessed 2026-01-08. 2026.

[3] 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.

[4] Z. Guo, X. Zhang, H. Mu, W. Heng, Z. Liu, Y. Wei, and J. Sun. "Single path one-shot neural architecture search with uniform sampling." In: *Computer vision – ECCV 2020*. Ed. by A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm. Cham: Springer International Publishing, 2020, pp. 544–560. ISBN: 978-3-030-58517-4.

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

[6] 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].

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

[8] 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.

[9] 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]`.

[10] 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`.

[11] 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`.

[12] 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.

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

[14] 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`.

[15] 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`.