Spraakwaterval

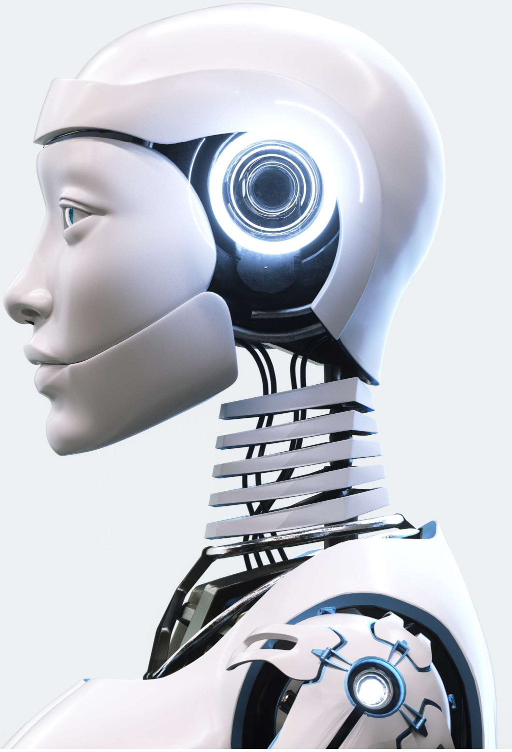
Fine tuning a language model in a federated setting

E. de Graaf, L. Tealdi

Start presentation

Agenda

1. Introduction
2. PET and Federated Learning
3. Tools:
   a. Data set: common voice
   b. Whisper and Hugging face
   c. Flower
4. Experiment
5. Next steps



Spraakwaterval – Introduction

Spraakwaterval is a collaboration between CZ, NPO and TNO.

It ran from November 2022 to April 2023

Contacts:

- Project manager: Maljaars, C.E.P. (Lizette) lizette.maljaars@tno.nl
- POC: Tealdi, L. (Lucia) lucia.tealdi@tno.nl; Graaf, E.W. (Erik) de erik.degraaf@tno.nl

Spraakwaterval – Introduction

Automatic speech recognition (ASR) systems can **enable machines to respond to human voice**, usually converting speech into text. As such they play a key role in automation and development of human-machine interaction and collaboration.

Possible use-cases of ASR tools:

- Customer services
- Automated transcription of conversations, meetings, TV shows etc

Last generation ASR systems are **AI based**, and are usually composed of **deep neural networks**, that require **huge quantity of good quality data** to be trained.

Spraakwaterval – Introduction

CHALLENGES

- Dataset are scattered across multiple organizations
- Legal and confidentiality issues prohibit sharing data

- ASR models need to be inclusive and work for different kind of voices, accents, minorities:
  - How to collect diverse datasets
  - How to deal with biases that may be present in speech data

- Dutch language ASR are currently limited because of limited availability of Dutch speech (labelled) datasets

AIM

Develop a Proof-of-Concept to demonstrates

how an aggregated model can be trained on multiple, distributed Dutch speech datasets

while maintaining confidentiality,

and providing insight in how to quantify bias in speech datasets.

DELIVERABLES

1. Slide deck: literature overview of potential PET solutions and their (dis)advantages to combine speech data

2. Slide deck with an overview of privacy sensitive properties of speech data

3. Slide deck with leads on how to quantify bias in speech data

4. **PoC to demonstrate a technical solution to share speech data**
5. **Report describing the outcome of deliverables 1 – 4**

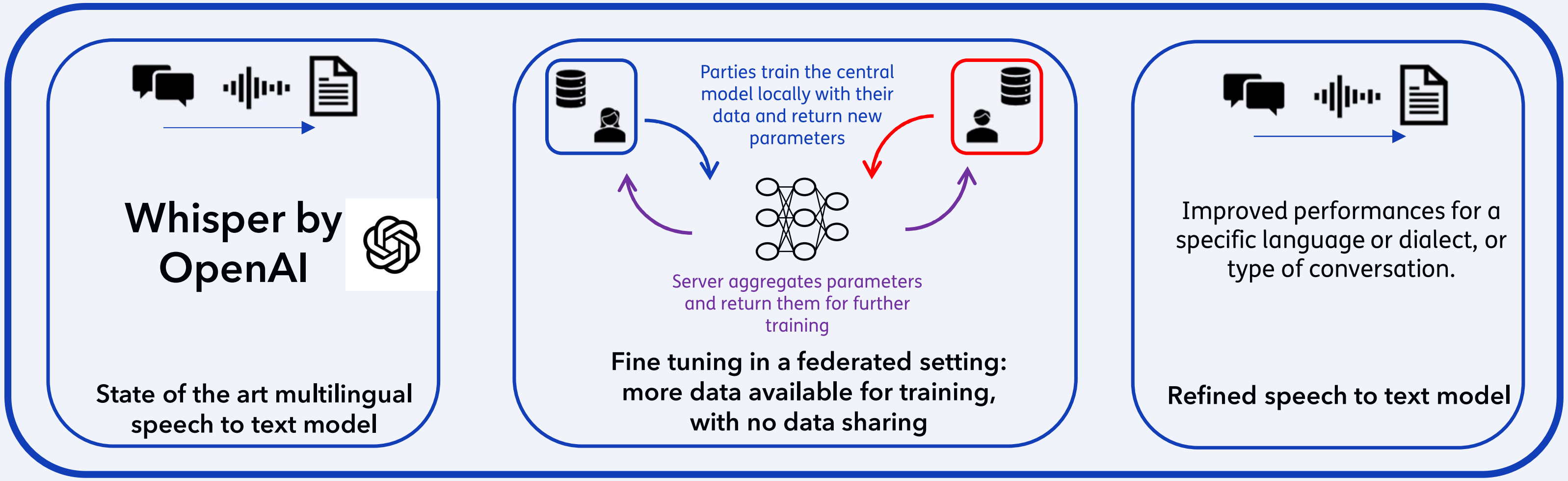
This slide deck should be considered as deliverable 5, covering the results related to deliverable 4. Deliverables 1 – 3 have been reported separately.

Spraakwaterval - POC

How an aggregated model can be trained on multiple, distributed Dutch speech datasets ?

PLAN

Demonstrate a possible use of a privacy enhancing technology called Federated Learning to facilitate the collaboration of different parties to co-fine-tune a ML model using different datasets that are, in this way, never shared.



Whisper by OpenAI

State of the art multilingual speech to text model

Parties train the central model locally with their data and return new parameters

Server aggregates parameters and return them for further training

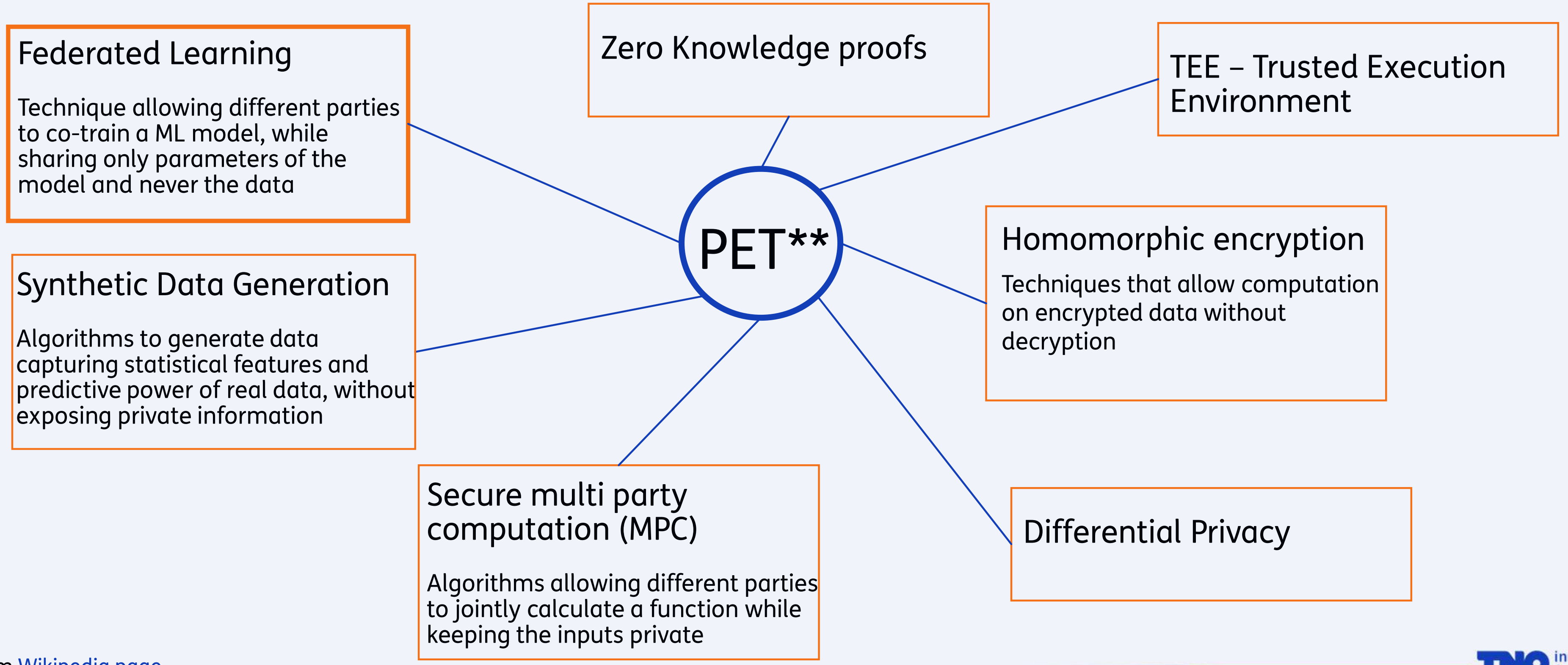
Fine tuning in a federated setting: more data available for training, with no data sharing

Improved performances for a specific language or dialect, or type of conversation.

Refined speech to text model

Privacy enhancing technologies (PET)

*Technologies that embody fundamental data protection principles by minimizing personal data use, maximizing data security, and empowering individuals**



PET**

Federated Learning

Technique allowing different parties to co-train a ML model, while sharing only parameters of the model and never the data

Zero Knowledge proofs

TEE – Trusted Execution Environment

Homomorphic encryption

Techniques that allow computation on encrypted data without decryption

Synthetic Data Generation

Algorithms to generate data capturing statistical features and predictive power of real data, without exposing private information

Secure multi party computation (MPC)

Algorithms allowing different parties to jointly calculate a function while keeping the inputs private

Differential Privacy

* From Wikipedia page

** Non-exhaustive list. Longer description is provided for which TNO has acquired large experience in the last years.

Privacy enhancing technologies (PET)

*Technologies that embody fundamental data protection principles by minimizing personal data use, maximizing data security, and empowering individuals**

Different technologies offer different levels of

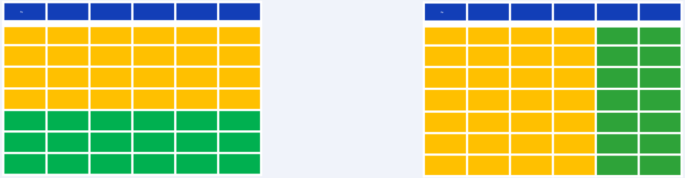
- Data protection
- Privacy guarantees
- Security
- Utility
- Computational efficiency

Federated learning offers in general less privacy guarantees than other techniques (e.g. homomorphic encryption or mpc) but it is more efficient and scalable. For a detailed discussion see deliverable 1.

Federated Learning

- First introduced by Google in 2017*
- Use case: several parties own different data that can contribute to train a ML model. FL allows to co-train the model without sharing the data
- Different cases are identified depending on how the data is partitioned:
  - Horizontal federated learning: parties own data with the same structure about different individuals/events
  - Vertical federated learning: parties own different features of the same individuals/events

  In this project data is horizontally distributed.



Visualization of horizontally (left) and vertically (right) distributed data. Orange and green areas represent the data owned by party 1 and 2 respectively.

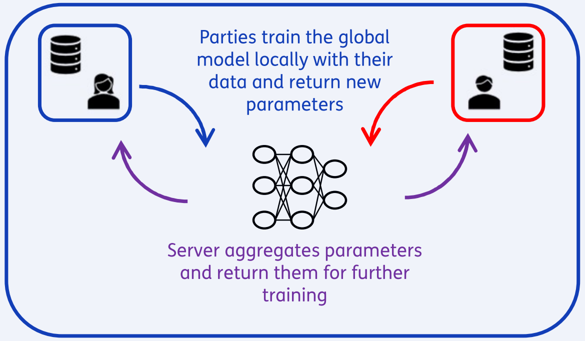
- In FL a model is trained in several iterations:

  0- All parties share an initial global model

  1- All parties train a local model using their data

  2- All the local parameters are aggregated (either in a central trusted party, or in a peer-to-peer fashion) to create an updated global model

  1 and 2 are repeated till when the model converges or the utility is satisfying



Parties train the global model locally with their data and return new parameters

Server aggregates parameters and return them for further training

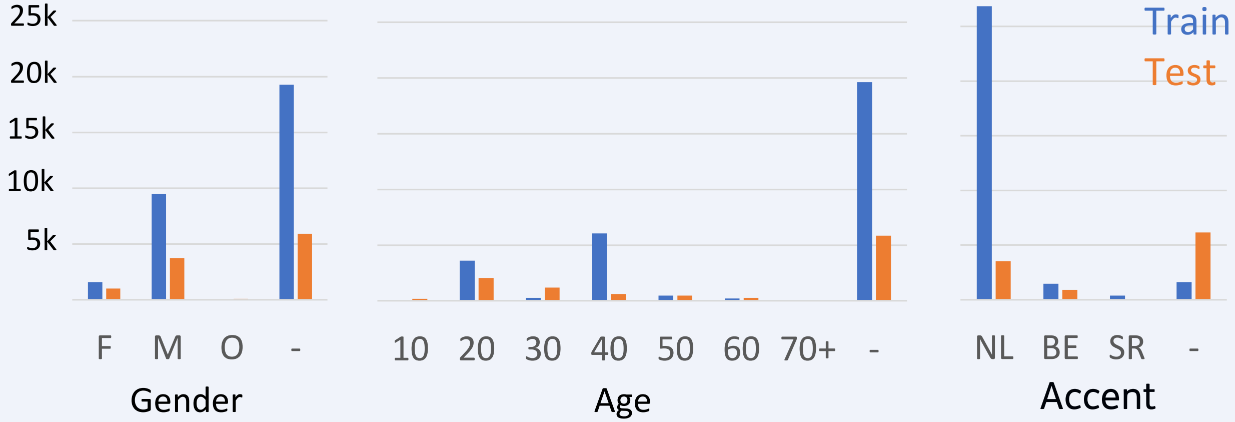
Scheme of ML training in federated setting

* McMahan, Brendan, et al. "Communication-efficient learning of deep networks from decentralized data." *Artificial intelligence and statistics*. PMLR, 2017.

Data

- In order to fine tune a speech-to-text model we need labelled (i.e. transcribed) speech data.
- We have not been able to use real data from CZ and NPO because the legal ground to use the data in the experiment has been built in parallel with the POC
- We have therefore used the Common Voice Corpus 11.0 Dataset, published by Mozilla.

Validated hours	102
Number of voices	1530
Train dataset dim	30318
Test dataset dim	10743
Sensitive features	Age, Gender, Accent



The Dutch common voice dataset contains 105 hours of validated and transcribed speech data. For a part of these files metadata is also available about the age, gender and accent of the speaker. Gender can be female (F), Male (M), or other; accent distinguishes among Dutch from the Netherlands (NL), Belgium (BE) or Suriname (SR). The '-' indicates that the metadata was not filled in.

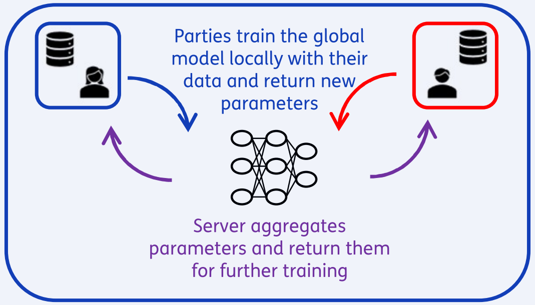
Language model: Whisper and Hugging Face

- Creating from scratch a language model is a very demanding task, especially for the amount of data that would be required. That is why we decided to fine-tune (i.e. further train with additional data) an existing ASR model.
- Whisper is a family of multilingual models developed by OpenAI*, and one of the state of the art ASR models. Different models are available, from whisper-tiny to whisper-large that differ in model dimension (i.e. number of model parameters).
- Hugging Face is a private American company developing libraries and applications using machine learning. It is especially known for the transformers library, that offers tools for Natural Language Processing applications**.
- Among others, Hugging face offers also a Transformer to interact with the Whisper model.

* Whisper white paper, 2022

** https://huggingface.co/docs/transformers/index

Federated Learning framework: flower


Parties train the global model locally with their data and return new parameters
Server aggregates parameters and return them for further training

Flower* is an open source library that provides a framework for federated learning.

In a nutshell it provides:

- A client basic class, that can be specialized to the ad-hoc local training
- A server basic class, that can be specialized adding ad-hoc aggregation functions, and other functionalities (e.g. centralized evaluation, saving models, ..)
- The implementation of the communication between parties (clients) and server.

We have chosen this framework because it offers a quick start, simulating different parties on the same machines (different prompts, or different docker containers), but can also be proficiently used in production environment.

We have not implemented the POC in a real multi-machine scenarios, since this was out of the project scope.

The technical requirements to guarantee the FL Flower orchestration are:

- Parties firewall allows large files over gRPC
- No TLS termination in the firewall

Note: The communication is NEVER about the raw data but 'only' about model parameters.

* https://flower.dev/

Experiments setup

Steps towards the POC:

- Fine tuning whisper-tiny* in a central setting (only one party training the entire dataset):

  Goal is to prove that we can actually improve the existing model further tuning it with the common voice dataset.

- Fine tuning whisper-tiny using only 25% - 50% - 75% of the training data set in a central way:

  Goal was to get a feeling of the importance of using as much as possible data.

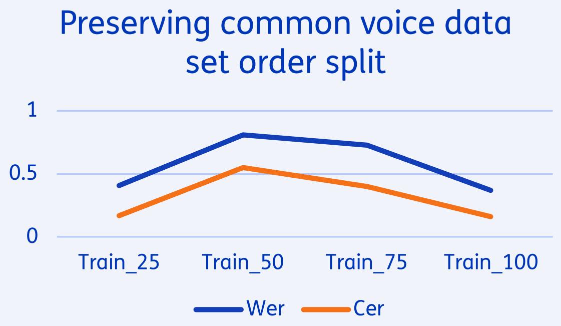
- Fine tuning whisper tiny in a federated setting:
  - We implemented a scenario with two parties
  - Data has been divided in two equal parts, keeping samples from the same voice together; this seemed closer to a real situation in which one party might have more samples from the same person (e.g. if several customer services collaborate)

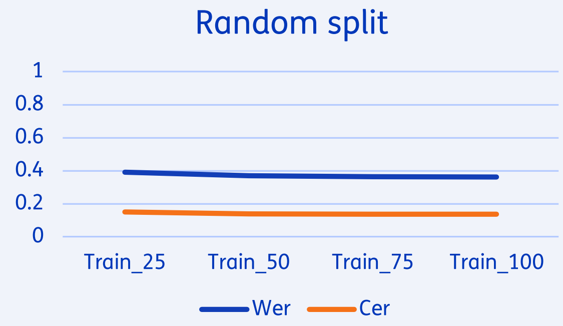
* whisper-tiny has been used in place of whisper-large because of time and memory constraints.
See next slides for more details.

MC(0

Experiments results: performances with more data

We split the training data in 4 parts, and trained (centrally) models using 25%, 50%, 75% and 100% of the data respectively. Results in term of word error rate (WER) and character error rate (CER) look very different depending on how we split the data:



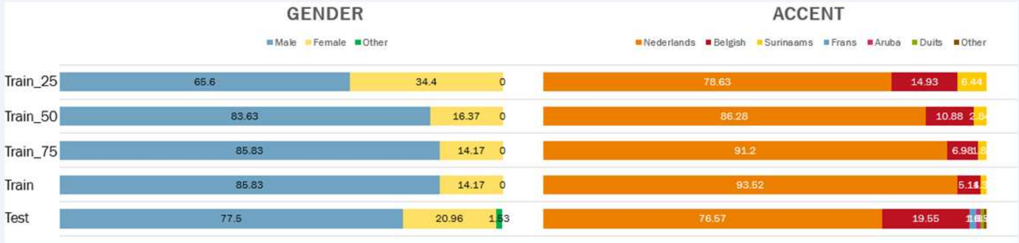


$$WER = \frac{S_w + D_w + I_w}{N_w}$$

$$CER = \frac{S_c + D_c + I_c}{N_c}$$

$S_{w,c}$ = substitutions
$D_{w,c}$ = deletions
$I_{w,c}$ = insertions
$N_{w,c}$ = total

The weird shape of the first graph is probably due to ‘bad luck’. The distribution of gender and accent seems more similar to the test dataset in the first quarter of data than in the rest.



Conclusion

More data → better results is a reasonable assumption but not a guarantee

Slide 14

MC(0 Needs definition of WER and CER

Maljaars, C.E.P. (Lizette); 2023-05-08T12:05:28.104

TL(0 0 Yes. I swaped this slide and the next one and forgot to repeat the definition. I decided to leave it in both slides for clarity

Tealdi, L. (Lucia); 2023-05-08T13:58:13.662

MC(0

Experiments results: central vs federated fine tuning

Models have been evaluated on the common voice test dataset, using two common metrics: word error rate (WER), and character error rate (CER)

Model	WER	CER
Whisper tiny	1,44	0,91
Centrally fine-tuned After 1000 steps	0,36	0,14
Federated fine-tuned *After 10 rounds of 100 steps each*	*1,12*	*0,60*

$$WER = \frac{S_w + D_w + I_w}{N_w}$$

$$CER = \frac{S_c + D_c + I_c}{N_c}$$

$S_{w,c}$ = substitutions
$D_{w,c}$ = deletions
$I_{w,c}$ = insertions
$N_{w,c}$ = total

- The federated fine tuned model outperforms the original whisper tiny model, but is clearly worse than the centrally fine tuned model (even considering models trained with less data).
- It would be interesting to experiment more with different split of data, and FL settings (e.g. how often the local models are aggregated). We expect there is room to get better results.
- WER and CER have been calculated transcribing the test dataset only once. Anyhow the model is not fully deterministic when transcribing; a better estimation of the metrics would require therefore multiple transcription and an error bar; it has not been done yet due to time constraints (see next slide).
- In the central setting, we fine tuned the model up to 5000 steps; the metrics further improve, but not drastically; 1000/2000 steps seem a reasonable amount.

Slide 15

MC(0 3rd bullet: This raises the question why this wasn't done. If I remember correctly, it was about the long computational time required for an experiment. Maybe good to add a footnote (or reference to the next slide) to proactively answer that question.

Maljaars, C.E.P. (Lizette); 2023-05-08T12:07:15.233

TL(0 0 done

Tealdi, L. (Lucia); 2023-05-08T13:57:35.489

Experiments: computational time and memory

Whisper-tiny has been used in place of whisper-large because of time and memory constraints.

On a server with GPU:

- Fine tuning: 8GB VRAM for Whisper Tiny, >24 GB VRAM for Whisper Large (our server could not handle it, that's why we have only a lower bound)
- Transcribing: 8GB VRAM for Whisper Large
- Training Whisper tiny takes about 16 hours for 5000 steps on 4 x Geforce 3090 in centralized mode
- 9 hours for 1000 steps in federated mode

Conclusions and lessons learned

- Federated Learning can help in fine tuning speech model with more data than what it might be available in a single dataset
- In our experiment performances are not competitive with those of a centrally fine tuned model; we hope that results can be improved with different choices of training parameters since FL has been already proved competitive in many use cases and domains.
- Working with speech data and big linguistic models takes time and computational resources.
- It makes also somehow difficult to evaluate and improve the development: selecting few of the speeches resulting in poor transcription suggests that they had a bad quality (background noise, even more than one voice speaking); the nature and dimension of the data did not allow a structural analysis within the boundary conditions of this project
- Depending on the use case, a centrally fine tuned model with less data might work better than a federated fine tuned model with lot of data: a model that really ‘fits’ a certain type of data (a regional accent) might be more desirable than a better generic model.

Next steps

- More experiments in the federating setting with different splits of data, and different training parameters
- Fine tuning the large whisper model (or other language model) to measure the room for improvement having additional data
- Set up a multi-machine federated framework
- Have a privacy evaluation of the tool: FL prevents for sharing the raw data, but information about the training datasets might leak from the parameters updates. If needed FL can be combined with other PET techniques (e.g. MPC of homomorphic encryption) but this come with additional communication and computational costs.

Thank you

TNO innovation for life