ISMB2022 Tutorial Federated Learning in Biomedicine

Speakers

- Julian Matschinske, UHH*
- Julian Späth, UHH
- Niklas Probul, UHH
- Mohammad Bakhtiari, UHH

*University of Hamburg

Target audience

- Bioinformaticians
- Computational biologists
- Data scientists
- Medical informaticians

Programming skills (ideally Python) and past experience with machine learning are advised

Repository

The GitHub repository is available at https://github.com/FeatureCloud/ismb-tutorial-2022

Abstract

The vast amount of biomedical data produced by recent sequencing technologies has shown to be a valuable resource for machine learning models to better understand biological mechanisms and pathways. While machine learning models generally depend on centralized datasets, unfortunately, this is not suited for sensitive medical data, which is often distributed across different institutions and cannot be easily shared due to high privacy or security concerns.

Federated learning, a method proposed by Google in 2017, allows the training of machine learning models on geographically or legally divided datasets without sharing sensitive data. When combined with additional privacy-enhancing techniques, such as differential privacy or secure multi-party computation, it can serve as a privacy-aware alternative to central data collections while still enabling the training of machine learning models on the whole dataset. This is achieved by exchanging (possibly obfuscated) model parameters only. However, in such federated settings, both algorithms, as well as the required infrastructure, are much more complex than for centralized machine learning approaches. To address this, various federated learning tools have been developed and published recently that try to fill this gap and make the usage and development of federated algorithms easier, more intuitive, and applicable for data scientists without requiring profound software engineering capabilities.

In this tutorial, first, the theory of federated learning will be introduced using Python examples. The risk of privacy leaks is demonstrated to show the necessity of additional privacy-enhancing techniques, which are introduced afterward. The acquired knowledge will then be put to use with the help of two tools, namely PySyft and FeatureCloud. These tools allow for implementing and executing federated algorithms in a truly federated production setting and will be used to provide the attendants with a practical hands-on experience, involving a real-world biomedical dataset and prediction task.

In the end, this tutorial will provide the attendants with both theoretical and practical knowledge about federated learning and privacy-enhancing techniques in the context of biomedicine and demonstrate the whole development process from the conception of the algorithm to deployment to a production system.

Learning objectives

- Federated learning theory and hands-on experience
- Privacy-enhancing techniques (differential privacy, secure-multiparty computation)
- Tools to implement federated algorithms/methods (sklearn, FeatureCloud, PySyft)
- Deployment of federated algorithms/methods

After attending the tutorial, attendants should have a solid understanding of what federated learning is, how it can be used to perform privacy-aware machine learning on distributed datasets using the techniques mentioned above, and how to practically bring such implementations to the user.

Agenda

From	To	Title and brief description	Speaker(s)
9:00 am 16:00 CEST	**9:15 am** 16:15 CEST	**Welcome and introduction**	Julian Matschinske
9:15 am 16:15 CEST	**9:45 am** 16:45 CEST	**Federated learning in biomedicine** theory The basic concepts and pitfalls related to federated learning, as opposed to central machine learning are shown, taking a look at loss of accuracy and performance and how to mitigate it.	
9:45 am 16:45 CEST	**10:45 am** 17:45 CEST	**Privacy-enhancing techniques** theory Approaches such as differential privacy and secure multi-party computation are introduced and discussed.	Niklas Probul
10:45 am 17:45 CEST	**11:00 am** 18:00 CEST	☕ **Coffee break**	-
11:00 am 18:00 CEST	**12:00 pm** 19:00 CEST	**How to develop your federated method** hands-on Having the concept ready, attendants learn how to actually implement a federated method, and take care of communication and orchestration using PySyft and FeatureCloud.	Mohammad Bakhtiari
12:00 pm 19:00 CEST	**12:50 pm** 19:50 CEST	**How to run your federated method** hands-on Central methods can be installed and executed individually, not so federated algorithms. They require a more complex deployment process which is demonstrated in this session.	Julian Späth
12:50 pm 19:50 CEST	**1:00 pm** 20:00 CEST	**Wrap-up**	