This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826078.

Privacy preserving federated machine learning and blockchaining for reduced cyber risks in a world of distributed healthcare

Deliverable D4.5
"Framework for Local Sphere privacy-aware federated learning on graphs"

Work Package WP4
"Supervised Federated Machine Learning"

Disclaimer

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826078. Any dissemination of results reflects only the author's view and the European Commission is not responsible for any use that may be made of the information it contains.

Copyright message

Document information

Grant Agreement Number: 826078			**Acronym: FeatureCloud**	
Full title	Privacy preserving federated machine learning and blockchaining for reduced cyber risks in a world of distributed healthcare			
Topic	Toolkit for assessing and reducing cyber risks in hospitals and care centres to protect privacy/data/infrastructures			
Funding scheme	RIA - Research and Innovation action			
Start Date	1 January 2019	**Duration**	60 months	
Project URL	https://featurecloud.eu/			
EU Project Officer	Christos MARAMIS, Health and Digital Executive Agency (HaDEA) - Established by the European Commission, Unit HaDEA.A.3 – Health Research			
Project Coordinator	Jan BAUMBACH, UNIVERSITY OF HAMBURG (UHAM)			
Deliverable	D4.5 "Framework for Local Sphere privacy-aware federated learning on graphs"			
Work Package	WP4 "Supervised Federated Machine Learning"			
Date of Delivery	**Contractual**	31/12/2022 (M48)	**Actual**	19/12/2022 (M48)
Nature	Report	**Dissemination Level**	Public	
Lead Beneficiary	03 MUG			
Responsible Author(s)	Prof. Dr. Andreas Holzinger, MUG			
Keywords	Graph Neural Networks, Federated Machine Learning, Counterfactuals, Human-in-the-loop			

Table of Content

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

1 Objectives of the deliverable based on the Description of Action (DoA)

WP 4 will contribute to theoretical and experimental research, design and develop federated and interactive learning approaches, following the “privacy by design and architecture” with a focus on the human-in-the-loop. Additionally, WP 4 will experiment and evaluate Explainable AI (xAI) approaches in order to make machine learning results transparent, re-traceable, re-enactable, explainable, interpretable and eventually understandable and ultimately test and evaluate Human-AI interfaces for this purpose.

Both robustness and explainability are gaining more and more importance for the European Union (Hamon et al. (2021)), because it can help to build trust (Holzinger et al. (2022)) in the systems, as users are more likely to accept the output of a system when they understand how the result was obtained. Furthermore, explainability can help to identify and address any biases and/or errors in the system, and facilitates debugging and development by the expert. Explainability is particularly important in regulated industries or in situations where the consequences of a wrong decision can be severe - such as in the health domain.

Moreover, in Objective 4 we explore the possibility of creating “local spheres” enabling privacy-aware federated learning on graphs using ensemble techniques (Task 4). Ensembling involves the training of multiple models on different subsets of the data and their predictions are combined. This is especially useful for federated learning, because it allows each device or silo to retain control over its own data, while still contributing to the overall model. We employ the ensemble technique for federated learning as follows: We divide the data into "local spheres" or subsets and train a separate model on each local sphere. Once the models are created, the predictions are combined using majority voting, weighted averaging, or bootstrapping.

The Milestone MS27(“Local sphere framework ready”) refers to this Deliverable and was also achieved.

2 Executive Summary

- Methodology: A federated Graph Neural Networks (GNN) application was developed. Each client contains a GNN with the same architecture and starts with the same initial model weights for all clients. The graph topologies of each client may be the same in the beginning, but after a few actions from the user, they will start being completely different. The changed graphs could be compared afterwards with graph comparison metrics as well as the selected action sequences of the users since we currently track and store each of them. MUG implemented the federation following the principles of the FedAvg algorithm (McMahan et al. (2017), He et al. (2021)). There is a possibility that the starting datasets are of different forms with node and edge features drawn from different distributions, hence being non-independent and identically distributed (i.i.d.).
- Results: The results are all openly available in the following public GitHub repository https://github.com/asaranti/GNN_Counterfactuals which contains documentation and issue tracking and will be presented in the form of a publication. The results of deliverable D4.6 were studied by two users that experimented on the same dataset, driven and guided by the Explainable AI (xAI) methods, the model performance and their domain knowledge. Their insights and findings are also reported in Beinecke et al. (2022). The federated (global) model from those two users and their interactions contains the distilled knowledge and thought processes of both; it is built indirectly via the changes that they make each in their own

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

datasets and at the same time is dependent on the federation algorithm i.e., which aggregation function is selected or even adaptively computed.

- Distinctive features: The graph datasets of each client are not revealed to each other. This ensures the privacy of the dataset to a large extent. Several Python scripts were developed to simulate random sequences of actions on the datasets, as well as their correct tracking for transparency, reproducibility and re-building from scratch. Furthermore, structured sequences of actions that were guided from xAI relevance values - either in decreasing or increasing order - were also simulated. Even before deployment, the effects of the actions on the weights and their averaging can be largely known. The simulation of actions can help us in the development of federated models by quantifying their performance before exploring and implementing them with real users.
- Progress beyond the state-of-the-art: The federation does not necessarily have to happen based on a weighted average; even more when GNN architectures are based on the idea of message passing and learned aggregation function. Thereby, the aggregation can be chosen from a set of nonlinear functions or even be searched in parameter space with the use of Genetic Algorithms (GA) or even Reinforcement Learning (RL). The best strategy is not yet fully resolved within the literature. We anticipate novel insights from knowledge-based human interactions with the model to derive an optimised aggregation technique.
- Progress beyond the state-of-the-art (continued). First experiments on ensemble-based GNN aggregation were conducted (also relevant for Deliverable D4.7). The underlying approach decomposes the input graph into local subnetworks ("local spheres") using community detection algorithms and explainable AI (Pfeifer et al. (2022)). The subnetworks are computed using our developed Python package GNN-SubNet (https://github.com/pievos101/GNN-SubNet) or are detected by a human expert which interacts with the graphs. The inferred subnetworks are collected from all clients and final predictions are based on, for instance, majority voting. The difference to the aforementioned strategy is that actions and the resulting adaptations to the global model are shared only, and only if a user has determined a whole subnetwork ("local-sphere"). See https://github.com/pievos101/Ensemble-GNN for our developed algorithm for federated ensemble-learning approach.

3 Introduction (Challenge)

The core idea of the Federated Average (FedAvg) algorithm is to train a machine learning model on multiple de-centralized datasets, or "clients," without the need to share any raw data. This is achieved by training a model on each client's data locally and then averaging the model updates across all clients (details in Konecny et al. (2016)).

The FedAvg algorithm consists of the following steps:

1. Initialize the model: The model is initialized with some initial parameters.
2. Select a subset of clients: A subset of clients is randomly selected.
3. Train the model on the selected clients: The model is trained on the data of the selected clients using local stochastic gradient descent (SGD).
4. Average the model updates: The model updates from each client are averaged to obtain a global model update.
5. Update the model: The global model update is applied to the model to update its parameters.
6. Repeat: Steps 2-5 are repeated for a predetermined number of rounds or until the model has converged.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826078.

The FedAvg algorithm is a simple and effective way to train machine learning models on decentralized data, and it has been used in a wide range of applications, including recommendation systems, natural language processing, and computer vision.

The mathematical foundation of the FedAvg algorithm is summarized in the following equation:

$$\underset{\boldsymbol{W}}{min} F(\boldsymbol{W}) = \underset{\boldsymbol{W}}{min} \sum_{k=1}^{K} \frac{N(k)}{N} f^{(k)}(\boldsymbol{W})$$

The minimization of the loss function *F(**W**)* of the central GNN model with weights ***W*** is considered to be equivalent to the minimization of the losses of each of the local GNN models f^(k)(***W***) with the same weights ***W*** as the central one, weighted by their relative number of samples. The only entities that are exchanged are the GNN’s weights.

The important research question here is at what stage the averaging of the model weights should be triggered. For this particular deliverable, we have investigated strategies to infer high performing subnetworks (“local spheres”) (Pfeifer et al. (2022)), which are determined locally, algorithmically or by human expert actions, and from each client independently. The inferred subnetworks are finally used to trigger the aggregation, via ensembling or averaging of the model weights.

With this approach “experimental” and “unnecessary” changes to the global model are avoided, which overall may lead to a more robust classifier.

4 Methodology

Federated learning allows models to be trained on decentralized data, where the data is distributed across multiple devices or silos. This can be useful in situations where the data is sensitive or private, and cannot be shared with a centralized server.

One way to ensure privacy-aware federated learning on graphs is to use techniques such as differential privacy, which is a method of adding noise to the data in order to protect the privacy of individual data points. Other methods for ensuring privacy in federated learning include using secure multi-party computation or homomorphic encryption to perform calculations on encrypted data. It's also important to consider the overall architecture and design of the federated learning system, including the communication protocols and the roles and responsibilities of the different parties involved.

We experimented with several such approaches and went beyond-state-of-the-art by integrating a human-in-the-loop principle (refer also to Holzinger et al. (2021)), where actions are not instantly used to update the global model, but instead only if an important subnetwork (“local sphere”) is detected. A local sphere refers to the subset of data or nodes that are used to train our model on a particular device or silo. In federated learning from graphs, the local sphere may correspond to a subgraph of the overall graph structure, containing a set of nodes and edges that are relevant to the specific device or silo.
The local sphere may be determined based on various factors such as the characteristics of the data or the task being performed. For example, in a federated learning scenario involving bio-medical

data, the local sphere for a particular device may include the data from a specific hospital or clinic or any other sensitive source. In general, the goal of federated learning is to train a global model that can make accurate predictions or decisions based on the combined data from all of the devices, while still respecting the privacy and autonomy of each device.

In a human-in-the-loop approach, a human expert (e.g., medical doctor, biologist, health specialist etc.) is involved directly into the machine learning pipeline, either by providing input or by reviewing and approving the output of a machine learning model. This can be useful in situations where it is important to ensure the robustness and explainability of the model's predictions or decisions. Sometimes (not always of course) the human expert can bring in knowledge which is otherwise not accessible to any machine learning algorithm.

To support both robustness and explainability in a human-in-the-loop approach, the human should be trained or knowledgeable about the domain and the machine learning process, so as to understand and interpret the model's output. The human should also be able to identify and diagnose any potential issues or biases in the model, and should be able to provide feedback or additional input as needed.

One way that the human can support robustness is by reviewing and verifying the model's output, and by providing additional data or context when necessary. For example, if the model makes a prediction that is not consistent with the human's knowledge or experience, the human can provide additional information or feedback to help the model improve.

To support explainability, the human should be able to understand the reasoning behind the model's predictions or decisions, and should be able to communicate this to others who may be interested in understanding how the model works.

This may involve reviewing the model's features and weights, or examining the data and algorithms used to train the model. The human may also need to provide explanations or justifications for the model's output to stakeholders or decision-makers.

We focused on the human-in-the-loop approach already in this task, because we need this approach in the further task as well. In the following figure our principle of federated human-in-the-loop learning, is presented:

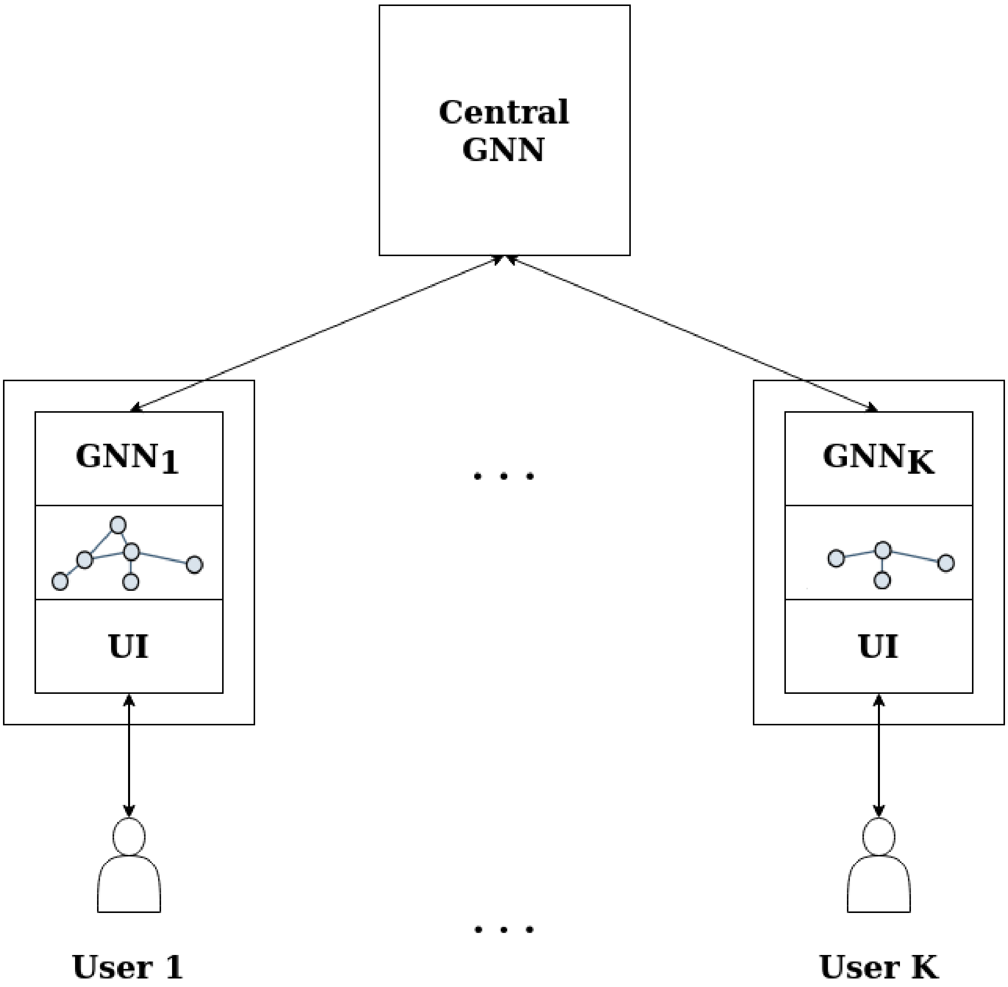

Figure 1: Federated Human-in-the-loop Counterfactual xAI Platform. Each user has a local dataset, and local Graph Neural Networks (GNN) and interacts with the same locally deployed UI. The central GNN is created by the transferred components of the local GNNs, which can be weights and/or embeddings.

In order to go beyond the FedAvg algorithm, we have developed our own ensemble-based approach, which can be federated using a similar strategy as for random forests (Malle et al. (2017)).

The user (or algorithm) infers relevant subnetworks, which ultimately form the local ensemble classifier. GNN models are trained on the inferred subnetworks ("local spheres") and are shared with the other clients in order to form the global/central model. Local predictions are made using majority voting based on the aggregated model (see Figure 2).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826078.

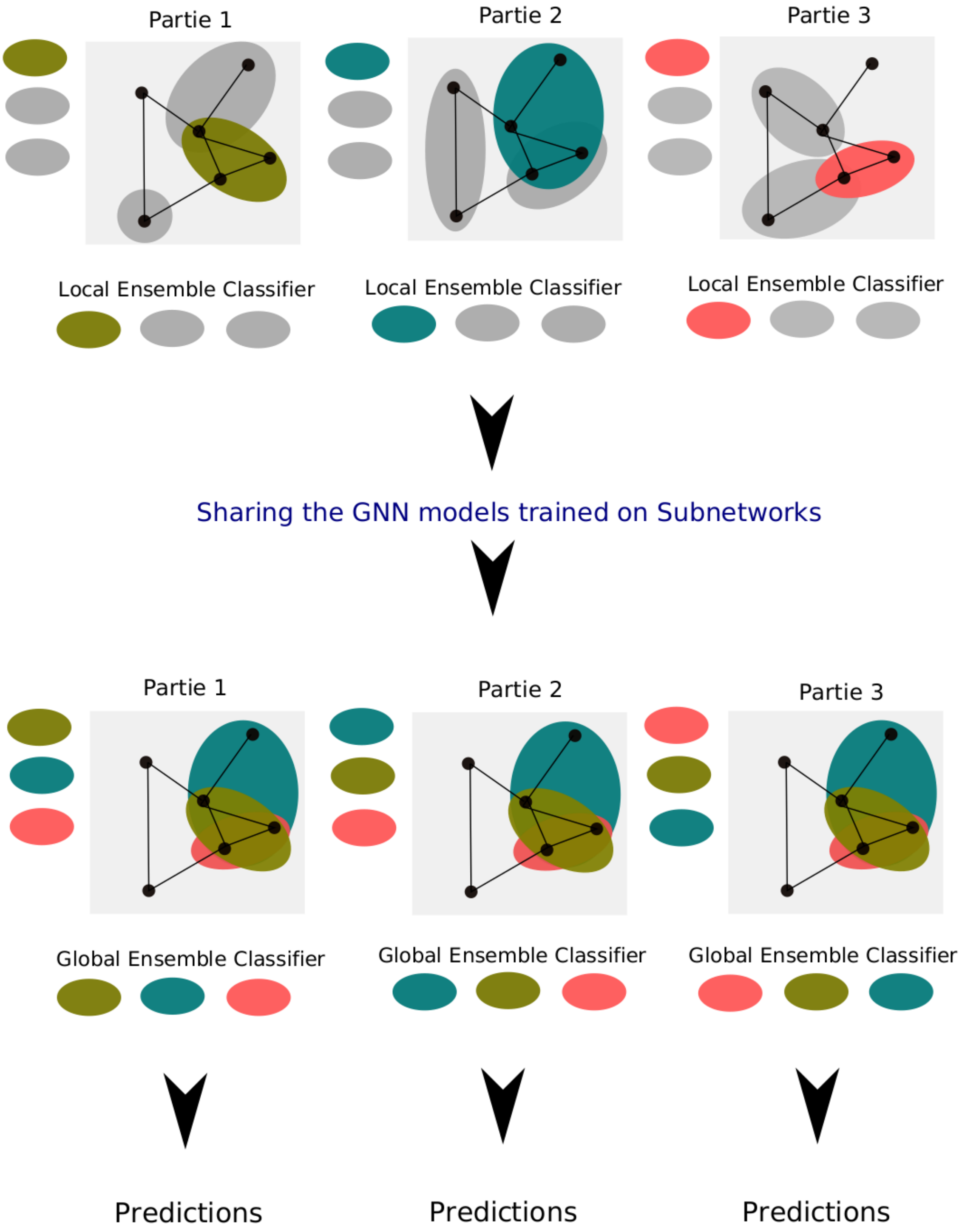
Partie 1
Partie 2
Partie 3
Local Ensemble Classifier
Local Ensemble Classifier
Local Ensemble Classifier
Sharing the GNN models trained on Subnetworks
Partie 1
Partie 2
Partie 3
Global Ensemble Classifier
Global Ensemble Classifier
Global Ensemble Classifier
Predictions
Predictions
Predictions

Figure 2: Each client builds its own ensemble classifier, algorithmically or due to human interactions. The ensemble classifier is based on the inferred subnetworks, so-called ‘local spheres’, which can be used to trigger the updating scheme of the global model.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

5 Results

We fulfilled the goals of this deliverable by developing the following pipeline for each client:

1. Each user interacts with the interface of one GNN and changes the dataset through graph actions (node add/delete, edge add/delete, features add/delete). Thereby, the datasets at the disposal of each client are different.
2. The local client’s GNN changes each time the user requires retraining. Unless all actions of a user were performed on graphs that belong to the test dataset, the local client’s GNN’s weights will change as a result of the retraining.
3. An asynchronous process uses the changed GNN to update the contents of the central GNN. The central GNN contains the knowledge of all user interactions.
   In the case of our “local sphere” approach, locally triggered *intermediate* interactions are not updating the central model. The “local spheres” are determined locally and the resulting weights are finally shared. Both approaches, however, need to be studied to finally infer the best possible strategy for federated learning on graphs, in terms of performance, privacy, and robustness.
4. For both approaches, the “central”/“global” GNN is to be tested on a dedicated dataset with different statistical properties than the ones of the client’s training sets.

6 Open issues

- Other averaging functions need to be tried out and if required, be adaptive.
- We are planning to test if an asynchronous update of the central GNN from the changed local ones is better, or a periodic update that does not take into account every client’s change. An asynchronous update means the local GNN’s weights are sent to the central model for aggregation and computation of the federated GNN after each local retrain. A periodic update computes the central federated GNN by taking the weights of all local GNNs at periodic time points, regardless of when retrain(s) were initiated. In this context, the ensemble-based approach serves as a baseline.
- So far, the “retrain” action is changing the whole GNN and retraining it from scratch. An online-retraining procedure, where the retrain would only slightly change the GNN’s weights is to be simulated.

7 Deviations (if applicable)

No deviations.

8 Conclusion

The federation of the GNN models of each client has been implemented. The central GNN model is the indirect result of human actions on the individual datasets, either by sharing all actions or by sharing only important subnetworks (Pfeifer et al. (2022)). Federation in GNN as we performed it during our work, refers to the process of training a model on multiple smaller graphs or subgraphs (“local spheres”), rather than training on a single large graph.

There are several advantages of our approach:

Scalability: Training on smaller subgraphs (“local spheres”) allows the model to be trained on larger graphs that may not fit in memory on a single machine (client-side computing is therefore possible, refer to Malle et al. (2017)).

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.

Efficiency: Training on smaller subgraphs allows the model to be trained faster, as the model can process each subgraph in parallel, refer to Malle et al. (2018).

Robustness: Training on multiple subgraphs can make the model more robust to noise or perturbations in the data, as the model can average out any errors that may occur on individual subgraphs.

Privacy: Federated learning allows the model to be trained on multiple de-centralized datasets without the need to share sensitive data between parties. This can be useful for applications where data privacy is a concern as in the medical domain.

Overall, federation in GNNs can help improve the scalability, efficiency, robustness, and privacy of graph learning tasks and needs much further work in the future.

9 References

McMahan, B. et al. (2017) ‘Communication-efficient learning of deep networks from decentralized data Artificial intelligence and statistics’, PMLR, pp. 1273–1282.

Hamon, R., Junklewitz, H. & Sanche, I. 2020. Robustness and Explainability of Artificial Intelligence - From technical to policy solutions, Luxembourg, Publications Office of the European Union, https://doi.org/10.2760/57493.

He, C. et al. (2021) ‘Fedgraphnn: A federated learning benchmark system for graph neural networks’, ICLR 2021 Workshop on Distributed and Private Machine Learning (DPML).

Holzinger, et al. (2021) ‘Towards Multi-Modal Causability with Graph Neural Networks enabling Information Fusion for explainable AI’. Information Fusion, 71, (7), 28-37, https://doi.org/10.1016/j.inffus.2021.01.008

Holzinger, A., Dehmer, M., Emmert-Streib, F., Cucchiara, R., Augenstein, I., Del Ser, J., Samek, W., Jurisica, I. & Díaz-Rodríguez, N. (2022). ‘Information fusion as an integrative cross-cutting enabler to achieve robust, explainable, and trustworthy medical artificial intelligence’. Information Fusion, 79, (3), 263--278, https://doi.org/10.1016/j.inffus.2021.10.007

Konečný, J., et al. (2016). ‘Federated learning: Strategies for improving communication efficiency’. arXiv:1610.05492.

Malle, B. et al. (2017) ‘The more the merrier -- federated learning from local sphere recommendations.”, International Cross-Domain Conference for Machine Learning and Knowledge Extraction. Springer, Cham. Available at: https://doi.org/10.1007/978-3-319-66808-6_24.

Malle, B., et al. (2018) ‘The Need for Speed of AI Applications: Performance Comparison of Native vs. Browser-based Algorithm Implementations’ arXiv:1802.03707.

Pfeifer, B. et al. (2022) ‘GNN-SubNet: Disease subnetwork detection with explainable graph neural networks.’, *Bioinformatics* 38.Supplement_2 (2022): ii120-ii126. Available at: https://doi.org/10.1093/bioinformatics/btac478.

Beinecke, J. et al. (2022) ‘Interactive explainable AI platform for graph neural networks’, bioRxiv. Available at: https://doi.org/10.1101/2022.11.21.517358.

10 Table of acronyms and definitions

AI	Artificial Intelligence
concentris	concentris research management GmbH
GA	Genetic Algorithm
GND	Gnome Design SRL
GNN	Graph Neural Network
MS	Milestone
MUG	Medizinische Universitaet Graz
RI	Research Institute AG & Co KG
RL	Reinforcement Learning
SDU	Syddansk Universitet
UHAM	University of Hamburg
UMG	University Medical Center Göttingen
UMR	Philipps Universitaet Marburg
WP	Work package
xAI	Explainable Artificial Intelligence

11 Other supporting documents / figures / tables (if applicable)

n/a

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 826078.