Bioinformatics, 2023, **39(11)**, btad703
https://doi.org/10.1093/bioinformatics/btad703
Advance Access Publication Date: 21 November 2023
Applications Note

OXFORD

Systems biology

Ensemble-GNN: federated ensemble learning with graph neural networks for disease module discovery and classification

Bastian Pfeifer [1,*,†], Hryhorii Chereda [2,†], Roman Martin [3], Anna Saranti [1,4], Sandra Clemens [3], Anne-Christin Hauschild[5], Tim Beißbarth[2], Andreas Holzinger [1,4], Dominik Heider[3]

[1]Institute for Medical Informatics, Statistics and Documentation, Medical University Graz, Graz 8036, Austria
[2]Medical Bioinformatics, University Medical Center Göttingen, Göttingen 37077, Germany
[3]Data Science in Biomedicine, Department of Mathematics and Computer Science, University of Marburg, Marburg 35043, Germany
[4]Human-Centered AI Lab, University of Natural Resources and Life Sciences, Vienna 1190, Austria
[5]Institute for Medical Informatics, University Medical Center Göttingen, Göttingen 37075, Germany

*Corresponding author. Institute for Medical Informatics, Statistics and Documentation, Medical University Graz, Auenbruggerplatz 2/9/V, Graz 8036, Austria. E-mail: bastian.pfeifer@medunigraz.at

[†]= equal contribution

Associate Editor: Janet Kelso

Abstract

Summary: Federated learning enables collaboration in medicine, where data is scattered across multiple centers without the need to aggregate the data in a central cloud. While, in general, machine learning models can be applied to a wide range of data types, graph neural networks (GNNs) are particularly developed for graphs, which are very common in the biomedical domain. For instance, a patient can be represented by a protein–protein interaction (PPI) network where the nodes contain the patient-specific omics features. Here, we present our Ensemble-GNN software package, which can be used to deploy federated, ensemble-based GNNs in Python. Ensemble-GNN allows to quickly build predictive models utilizing PPI networks consisting of various node features such as gene expression and/or DNA methylation. We exemplary show the results from a public dataset of 981 patients and 8469 genes from the Cancer Genome Atlas (TCGA).

Availability and implementation: The source code is available at https://github.com/pievos101/Ensemble-GNN, and the data at Zenodo (DOI: 10.5281/zenodo.8305122).

1 Introduction

Machine learning and deep learning offer new opportunities to transform healthcare, and have been used in many different areas, including oncology (Bibault *et al.* 2016), pathology (Coudray *et al.* 2018), diabetes (Spänig *et al.* 2019), or infectious diseases (Riemenschneider *et al.* 2016, Ren *et al.* 2022). However, clinical datasets are typically rather small and need to be aggregated over different hospitals. Often, data exchange over the internet is perceived as insurmountable, posing a roadblock hampering big data-based medical innovations. The most pressing problem in training powerful AI is that all the data usually distributed over various hospitals needs to be accessible in a central cloud. Due to recent data leaks, public opinion, and patient trust (Holzinger 2021, Holzinger *et al.* 2021) demand more secure ways of storing and processing data. As the E.U. GDPR (General Data Protection Regulations) and as the E.U. NISD (Network and Information Security Directive) entered into force in 2018 and 2016, respectively, data providers, researchers, and IT solution providers are challenged to find ways of providing hospitals complete control over how the patient data is processed. Federated AI enables collaborative AI without sharing the data and, thus, is a promising approach toward GDPR compliance. Federated AI implies that each participant securely stores its data locally and only shares some intermediate parameters computed on local data (Hauschild *et al.* 2022, Näher *et al.* 2023). It should be noted, that using federated learning alone does not automatically fulfill all GDPR requirements. For instance, federated learning does not protect against attacks such as model inversion.

Graph neural networks (GNNs) are widely adopted within the biomedical domain (Muzio *et al.* 2021). Biological entities such as proteins do not function independently and thus must be analyzed on a systems level. GNNs provide a convenient way to model such interactions using e.g prior knowledge defined by a protein–protein interaction network (PPI). The PPI can be used as the GNN's input graph, while the nodes can be enriched by patient-specific omics profiles. These knowledge-enriched deep learning models might be more interpretable compared to standard approaches when used to predict

Received: 13 April 2023; **Revised:** 6 September 2023; **Editorial Decision:** 10 November 2023; **Accepted:** 20 November 2023

patient outcomes. However, due to the GDPR mentioned above, the challenge remains how to enable federated learning on GNNs. Here, we present a software tool and Python package for federated ensemble-based learning with GNNs. The implemented methodology enables federated learning by decomposing the input graph into relevant subgraphs based on which multiple GNN models are trained. The trained models are then shared by multiple parties to form a global, federated ensemble-based deep learning classifier.

2 Materials and methods

2.1 Input data

The input data for our software package consists of patient omics data on a gene level and a PPI network reflecting the interaction of the associated proteins. In order to perform graph classification using GNNs, each patient is represented by a PPI network, and its nodes are enriched by the patient's individual omics features. We call these networks patient-specific PPI networks. Given that specific data representation, it is possible to classify patients based on their genomic characteristics while incorporating the knowledge about the functional relationships between proteins. It should be noted, that the topology of the network is the same for all patients.

2.2 Ensemble learning with graph neural networks

The proposed algorithm for graph-based ensemble learning consists of three steps:

1) Decomposition of the PPI network into communities using explainable AI.
2) Training of an ensemble GNN graph classifier based on the inferred communities.
3) Predictions via Majority Voting.

In the first step, the Python package GNNSubNet (Pfeifer *et al.* 2022) is used to build a GNN classifier and to infer relevant PPI network communities (disease subnetworks). In detail, GNNSubNet utilizes the Graph Isomorphism Network (GIN) (Xu *et al.* 2018) to derive a graph classification model and implements a modification of the GNNExplainer (Ying *et al.* 2019) program such that it computes model-wide explanations. This is done by randomly sampling patient-specific networks while optimizing a single-node mask. From this node mask, edge relevance scores are computed and assigned as edge weights to the PPI network. A weighted community detection algorithm finally infers disease subnetworks. In the second step, an ensemble classifier based on the inferred disease subnetworks is created, and predictions are accommodated via Majority Voting. The ensemble members are predictive GNN models, that are based on the detected disease subnetworks, which overall makes the deep learning model more interpretable. High performing members of the ensembles may consist of a subnetwork biologically important for a specific disease or disease subtype.

In the *federated* case, each client has its dedicated data based on which GNN models of the ensemble are trained. These models are shared among all clients creating a global ensemble model, and predictions are again accomplished via Majority Vote (see Fig. 1).

3 Results and discussion

We used the gene expression data of human breast cancer patient samples for an experimental evaluation of the herein proposed methodologies. The Cancer Genome Atlas (TCGA) provided the data preprocessed as described in (Chereda *et al.* 2021b). The data was structured by the topology of Human Protein Reference Database (HPRD) PPI network (Keshava Prasad *et al.* 2009). The resulting dataset comprises 981 patients and 8469 genes. The binary prediction task was to classify the samples into a group of patients with the luminal A subtype (499 samples) and patients with other breast cancer subtypes (482 samples).

3.1 Performance of Ensemble-GNN in nonfederated case

We assessed the performance of our algorithm using 10-fold cross-validation (see Table 1). Ensemble-GNN, initially using GIN as a base learner, showed the average balanced accuracy (Brodersen *et al.* 2010) of 0.86 (fourth row of Table 1). As a comparison, a Random Forest (RF) classifier, not guided and restricted by any PPI knowledge graph, demonstrated 0.90 of average balanced accuracy on the same dataset. The slight decrease in performance can be explained by the following reason: The GIN method (Xu *et al.* 2018) shows worse convergence during training on gene expression modality, compared to data with combined gene expression and DNA methylation modalities [see also Pfeifer *et al.* (2022), Table 2]. Since transcriptomics is one of the most common omics types (Athieniti and Spyrou 2023), we improved the performance of Ensemble-GNN specifically on gene expression modality using the ChebNet approach from Defferrard *et al.* (2016). ChebNet models have been successfully applied in Chereda *et al.* (2021a) to classify patients based on gene expression profiles structured by a PPI network. The corresponding predictions were further explained on a patient level (Chereda *et al.* 2021a). Ensemble-GNN employing ChebNet as a base learner achieved as good classification performance as RF (see Table 1).

For the GIN architecture we could observe that decomposing the PPI into subnetworks has a positive effect on the predictive performance. For the ChebNet architecture, however, no such effect can be reported. We explain this difference by the fact that the employed GIN classifier applies global pooling to the whole graph. In a classical graph classification setup pooling from an entire graph, comprising a small set of vertices, is more efficient than pooling from a graph with thousands of vertices. In our case the graph topology is the same across all patients, which might intensify the aforementioned effect. Furthermore, it has been reported that GNNs are susceptible to an effect called "oversmoothing" in multilayered architectures (Rusch *et al.* 2023). In our case, GIN had five message-passing layers while ChebNet comprised one graph convolutional layer and one fully connected layer, without global pooling. GNNs trained on smaller subnetworks could be less affected by the "oversmoothing" phenomenon. Our reported results on gene expression data using the GIN architecture support this observation; however, additional investigation is required to draw definitive conclusions.

3.2 Performance of Ensemble-GNN in federated case

In the federated case, we evaluated the performance of Ensemble-GNN using the two implemented base learners:

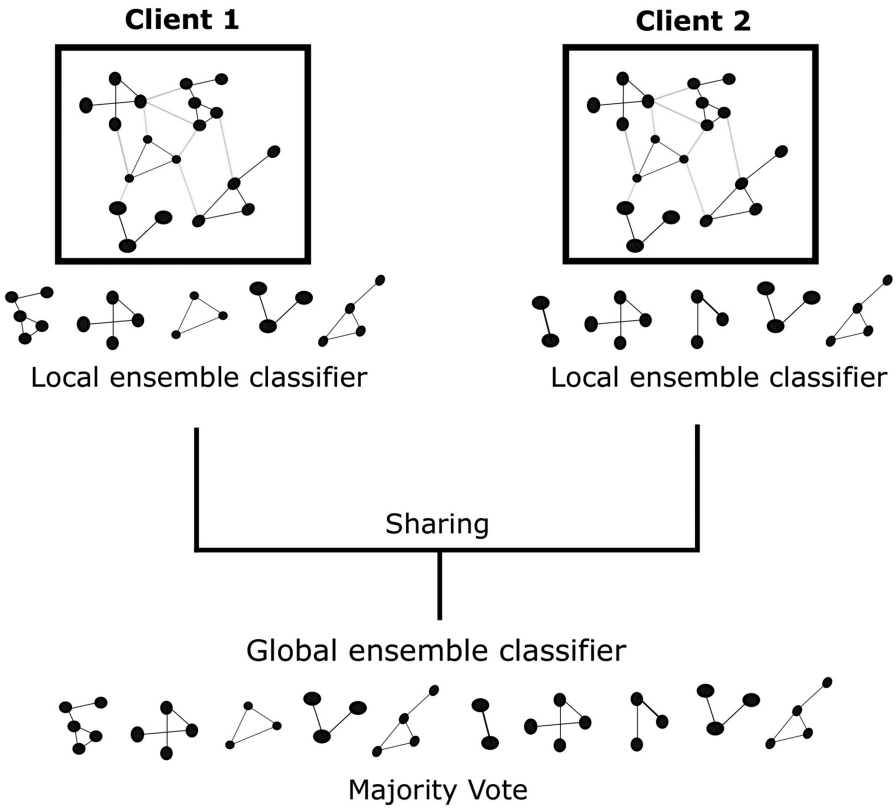

Figure 1. Federated ensemble learning with GNNs. Each client builds its dedicated ensemble classifier based on relevant subnetworks. The models trained on these subnetworks are shared and a global ensemble classifier is created. Final predictions are based on Majority Voting

Table 1. Performance within 10-fold CV (mean balanced accuracy ± standard error of the mean) of GNNSubNet and Ensemble-GNN utilizing TCGA-BRCA data, depending on a base learner (GIN or ChebNet).

Method (base learner)	Balanced accuracy
RF (decision tree)	0.90 ± 0.0109
GNNSubNet (ChebNet)	0.90 ± 0.0094
Ensemble-GNN (ChebNet)	0.90 ± 0.0105
Ensemble-GNN (GIN)	0.86 ± 0.0120
GNNSubNet (GIN)	0.52 ± 0.0121

GIN and ChebNet. We utilized five Monte Carlo iterations, in which the data was equally distributed across three clients. Each client-specific data was then split into a train (80%) and test dataset (20%). The trained models of the client-specific Ensemble-GNNs were combined into a global federated model. The performance of the global model was estimated using client-specific test sets.

For Ensemble-GNN (GIN), the mean local client-specific test accuracy from five Monte Carlo iterations was [0.84, 0.83, 0.78, 0.79, 0.81] with an overall mean value of 0.81 (see Table 2). For the global, federated model we obtained [0.85, 0.86, 0.83, 0.84, 0.87] with a mean value of 0.85. Mean performance values for Ensemble-GNN using the ChebNet architecture were higher, 0.85 and 0.87 respectively.

We have conducted an additional experiment, where we initially split the data into a global train set and a global test dataset. The train set was then equally distributed across three clients. The average performance of the local classifier to predict the global test set was [0.78, 0.80, 0.77, 0.77, 0.76] with an overall mean value of 0.78. The accuracy of the federated model was [0.82, 0.83, 0.82, 0.80, 0.86] with an overall mean value of 0.83. The performance of Ensemble-GNN could be improved using ChebNet. In this case, we obtained mean values of 0.87 and 0.88 respectively (see Table 2).

Note, we report on balanced accuracy in all cases to account for a possible unbalanced sample distribution caused by the data splits. However, whether the federated model can compensate for possible batch effects and/or non-IID (independent and identically distributed) data is uncertain. In our future work, we could use the concepts of federated domain adaptation as proposed by Peng *et al.* (2019) and transfer learning (Park *et al.* 2021) to make a global model adaptable to a specific client.

Finally, the overall accuracy could be further improved using aggregation techniques beyond Mayority Voting. For instance, the inferred subnetworks could be weighted by their relevances to obtain the final labels. Alternative voting rules were also developed and discussed by Werbin-Ofir *et al.* (2019).

4 Conclusion

We present Ensemble-GNN, a Python package for ensemble-based deep learning with interpretable disease subnetworks as ensemble members. The implemented methodology is especially suited, but not limited to, the federated case, where sensitive data is distributed across multiple locations. We could show that the models trained on subnetworks locally, and shared across multiple parties/clients globally, can improve client-specific predictive performance.

Table 2. Performance (mean balanced accuracy) of Ensemble-GNN based on TCGA-BRCA data, depending on the base learner GIN and ChebNet.[a]

Method (base learner)	Federated setup with client-specific test data		Federated setup with global test data	
	Local model	Federated model	Local model	Federated model
Ensemble-GNN (ChebNet)	0.85	0.87	0.87	0.88
Ensemble-GNN (GIN)	0.81	0.85	0.78	0.83

[a] For both federated setups, the data was distributed across three clients within five Monte-Carlo iterations.

Conflict of interest

None declared.

Funding

This work has received funding from the European Union's Horizon 2020 research and innovation programme [826078] (Feature Cloud). HC was supported by the German Ministry of Education and Research (BMBF) FAIrPaCT project [01KD2208A]. This publication reflects only the authors' view and the European Commission is not responsible for any use that may be made of the information it contains. Parts of this work have been funded by the Austrian Science Fund (FWF), Project: P-32554 explainable Artificial Intelligence.

Data availability

The Python package Ensemble-GNN, including comprehensive documentation of its usage is freely available from our GitHub repository (https://github.com/pievos101/Ensemble-GNN) as well as the data at Zenodo (DOI: 10.5281/zenodo.8305122).

References

Athieniti E, Spyrou GM. A guide to multi-omics data collection and integration for translational medicine. *Comput Struct Biotechnol J* 2023;**21**:134–49.

Bibault J-E, Giraud P, Burgun A. Big data and machine learning in radiation oncology: state of the art and future prospects. *Cancer Lett* 2016;**382**:110–7.

Brodersen KH, Ong CS, Stephan KE *et al.* The balanced accuracy and its posterior distribution. In: *2010 20th International Conference on Pattern Recognition*, Istanbul, Turkey. IEEE, 2010, 3121–3124.

Chereda H, Bleckmann A, Menck K *et al.* Explaining decisions of graph convolutional neural networks: patient-specific molecular subnetworks responsible for metastasis prediction in breast cancer. *Genome Med* 2021a;**13**:42.

Chereda H, Leha A, Beissbarth T. Stability of feature selection utilizing graph convolutional neural network and layer-wise relevance propagation. bioRxiv, https://doi.org/10.1101/2021.12.26.474194, 2021b, preprint: not peer reviewed.

Coudray N, Ocampo PS, Sakellaropoulos T *et al.* Classification and mutation prediction from non-small cell lung cancer histopathology images using deep learning. *Nat Med* 2018;**24**:1559–67.

Defferrard M, Bresson X, Vandergheynst P. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. arXiv, arXiv:1606.09375 [cs, stat], 2016, preprint: not peer reviewed.

Hauschild A-C, Lemanczyk M, Matschinske J *et al.* Federated random forests can improve local performance of predictive models for various health care applications. *Bioinformatics* 2022;**38**:2278–86.

Holzinger A. The next frontier: AI we can really trust. In: Kamp M (ed.), *Proceedings of the ECML PKDD 2021, CCIS 1524*. Bilbao, Spain: Springer Nature, 2021, 427–440.

Holzinger A, Dehmer M, Emmert-Streib F *et al.* Information fusion as an integrative cross-cutting enabler to achieve robust, explainable, and trustworthy medical artificial intelligence. *Inf Fusion* 2021;**79**:263–78.

Keshava Prasad TS, Goel R, Kandasamy K *et al.* Human protein reference database-2009 update. *Nucleic Acids Res* 2009;**37**:D767–72.

Muzio G, O'Bray L, Borgwardt K. Biological network analysis with deep learning. *Brief Bioinform* 2021;**22**:1515–30.

Näher A-F, Vorisek CN, Klopfenstein SAI *et al.* Secondary data for global health digitalization. *Lancet Digit Health* 2023;**5**:e93–101.

Park Y, Hauschild A-C, Heider D. Transfer learning compensates limited data, batch effects and technological heterogeneity in single-cell sequencing. *NAR Genom Bioinform* 2021;**3**:lqab104.

Peng X, Huang Z, Zhu Y *et al.* Federated adversarial domain adaptation. arXiv, arXiv:1911.02054 [cs.CV], 2019, preprint: not peer reviewed.

Pfeifer B, Saranti A, Holzinger A. GNN-SubNet: disease subnetwork detection with explainable graph neural networks. *Bioinformatics* 2022;**38**:ii120–6.

Ren Y, Chakraborty T, Doijad S *et al.* Prediction of antimicrobial resistance based on whole-genome sequencing and machine learning. *Bioinformatics* 2022;**38**:325–34.

Riemenschneider M, Hummel T, Heider D. SHIVA—a web application for drug resistance and tropism testing in HIV. *BMC Bioinformatics* 2016;**17**:314.

Rusch TK, Bronstein MM, Mishra S. A survey on oversmoothing in graph neural networks. arXiv, arXiv:2303.10993, 2023, preprint: not peer reviewed.

Spänig S, Emberger-Klein A, Sowa J-P *et al.* The virtual doctor: an interactive clinical-decision-support system based on deep learning for non-invasive prediction of diabetes. *Artif Intell Med* 2019;**100**:101706.

Werbin-Ofir H, Dery L, Shmueli E. Beyond majority: label ranking ensembles based on voting rules. *Expert Syst Appl* 2019;**136**:50–61.

Xu K, Hu W, Leskovec J *et al.* How powerful are graph neural networks? arXiv, arXiv:1810.00826, 2018, preprint: not peer reviewed.

Ying R, Bourgeois D, You J *et al.* Gnnexplainer: generating explanations for graph neural networks. *Adv Neural Inf Process Syst* 2019;**32**:9240–51.