Facing small and biased data dilemma in drug discovery with federated learning

Zhaoping Xiong[1,2,3#], Ziqiang Cheng[4,#], Xiaohong Liu[1,2], Dingyan Wang[2,3], Xiaomin Luo[2], Mingyue Zheng[*,2], Hualiang Jiang[*,1,2]

[1]*Shanghai Institute for Advanced Immunochemical Studies, and School of Life Science and Technology, ShanghaiTech University, Shanghai 200031, China;*
[2]*Drug Discovery and Design Center, State Key Laboratory of Drug Research, Shanghai Institute of Materia Medica, Chinese Academy of Sciences, 555 Zuchongzhi Road, Shanghai, 201203, China;*
[3]*University of Chinese Academy of Sciences, No.19A Yuquan Road, Beijing 100049, China*
[4]*School of Information Science and Technology, University of Science and Technology of China Hefei, 230000, China*
[#]*Authors contribute equally*

Abstract

Artificial intelligence (AI) models usually require large amounts of high quality training data, which is in striking contrast to the situation of small and biased data faced by current drug discovery pipelines. The concept of federated learning has been proposed to utilize distributed data from different sources without leaking sensitive information of these data. This emerging decentralized machine learning paradigm is expected to dramatically improve the success of AI-powered drug discovery. We here simulate the federated learning process with 7 aqueous solubility datasets from different sources, among which there are overlapping molecules with high or low biases in the recorded values. Beyond the benefit of gaining more data, we also demonstrate federated training has a regularization effect making it superior than centralized training on the pooled datasets with high biases. Further, federated model customization for each client can effectively help us deal with the highly biased data in drug discovery and achieve better generalization performance. Our work demonstrates the application of federated learning in predicting drug related properties, but also highlights its promising role in addressing the small data and biased data dilemma in drug discovery.

Introduction

Current artificial intelligence (AI) has high requirements for data both in terms of quality and quantity to achieve good predictive performance. Data acquisition difficulties and data biases in the measurement of scientific tests have significantly limited AI's power in drug discovery[1–3]. Data acquisition challenges come from time-consuming and expensive processes of data-generation and consequent confidentiality,

especially in the later stages drug development such as data about drug pharmacokinetics, safety, and efficacy profiles. Taking ADME/T (Absorption, Distribution, Metabolism, Excretion and Toxicity) properties as an example, such data are usually highly standardized and of good quality, and would contribute to better predictive model and generate larger added value when used for modeling. However, few of these properties are exposed to the latest deep learning models due to confidentiality[4,5], which can be considered as an enormous loss for drug development. Not only the data acquisition difficulties resulting from confidentiality, the data biases in the measurement of scientific tests also perplex AI for drug discovery[6]. It is common to see, for example, a specific molecular property has large discrepancy in recorded values from different sources, even under the same measurement of the same scientific tests. The discrepancy in recorded values is usually considered to come from data biases, because a recorded value in each data source is obtained from repeating measurements and the effect of variance has been reduced to minimum. In conventional machine learning paradigm, the discrepancy in different sources will usually be uniformed by taking the mean, median or a majority vote, which might bring the value closer to the "ground truth". However, for practitioners who generated those data may only care about the recorded value in their own experimental setting-ups for reference (i.e. whether the structure modification will lead to the optimization of a property), rather than the absolute "true" value. Therefore, a shared global model for all data sources might not form good instructions to practitioners. Federated learning emerges as a new machine learning paradigm provide viable solutions for data acquisition and data bias problems faced by AI drug discovery by keeping confidentiality and customizing model for users.

Federated learning represents a scenario where multiple clients can train a model collectively without sharing raw data[7–10]. The original idea dates back to 2016, in the context of the enactment of GDPR (General Data Protection Regulation) in Europe, users gain more control over the use of personal data, which challenges many companies that rely heavily on selling Ads based on users' personal data. McMahan et al. from Google proposed federated learning and a year later initially applied it in Gboard, the keyboard on android phones[7,8]. Therein, federated learning was adopted to train, for example, a next word prediction model cross many phone devices without uploading users' data to central servers. This process has improved users' input experience and preserved users' privacy. Noteworthily, aside from training a single global model collectively on clients' datasets, it becomes viable for each client to have a customized model in federated learning[9,10]. Therefore, the problem of discordant records in centralized machine learning turn into an intrinsic feature of federated learning[10]. Federated learning has attracted substantial attention and has found more and more applications in a much broader areas[13–17], which is also a promising approach to satisfy the needs of drug discovery[18] but yet to be investigated and tested. Drug discovery has similar request to protect the confidential or IP-sensitive data, and at the same time, to extract the maximum information/knowledge present within such data by machine learning. Moreover, given the high biases in drug discovery related data, customizing a model for each client is appealing for personalized prediction as done in

Gboard[8].

Here, we setup a general federated learning framework for drug discovery (Fig. 1) and tested it on FATE (Federated AI Technology Enabler)[19], an open-source project aiming at providing a secure computing framework for federated learning. Different from previously mentioned Gboard cross-device federated learning application that trained cross millions of phone devices, federated learning for drug discovery is another setting trained cross data silos, which is termed as cross-silo federated learning. In this setting, there are a coordinator server and several collaborators instrumented with federated learning client program. These clients in collaboration can be big pharmas, biotech startups or even academic labs having their own data silos.

During each round of cross-silo federated training, 1) the coordinator server broadcast the latest shared global model to each client, 2) each client locally computes the model updates by executing the training program, 3) and encrypts and uploads the model updates under a secure aggregation protocol, 4) finally, the coordinator server aggregates all the encrypted model updates securely and uses them to update the shared global model. Figure 1 illustrates the life cycle of federated learning system for drug discovery. Many rounds of training will be required until the model converges or meets the criteria for stopping, which might be the metrics that does not improve within given rounds on a shared dataset on the coordinator server or a held-out validation dataset on each clients. The best model is then selected for rollout. For users who want to use the model for prediction, they can use the selected shared model directly. Alternatively, for users who own plenty of labeled data themselves, they can opt to instrument the federated training program and locally update the model (without uploading updates to coordinator), thus obtaining a customized model. Model customization is a common application and should be very practically useful.

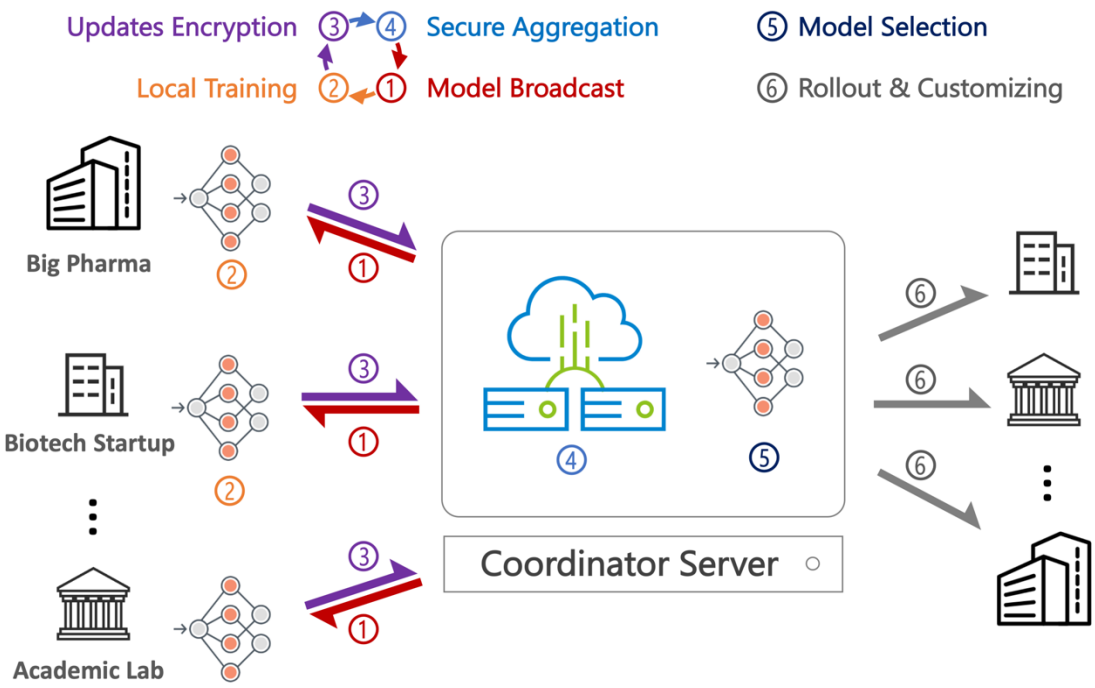

Fig. 1. The life cycle of a federated learning system for drug discovery. In federated

training: 1) the coordinator server broadcast the latest shared global model to each client; 2) the client locally computes the model updates, 3) encrypts and uploads the model updates; 4) finally, the coordinator server aggregates all the encrypted model updates securely and uses them to update the shared global model for the next round of training. After the training is done, the best model is selected for rollout and might be customized for the users who have their own labeled data.

In this work, we simulated a cross-silo federated learning process with seven existing solubility datasets that show variance in the chemical space of compounds covered, measurement methods, experimental conditions, nonstandard representations and size of data. These real-world drug property datasets from different sources represent non-identical data distributions at different clients, from which we would like to investigate how drug discovery projects can benefit from federated learning.

Results

Facing non-IID data

Table 1. Data statistics and the mean absolute deviation of the values for the compounds shared among different sources

F1	**6110 (0.00)**						
F2	763 (0.27)	**4650 (0.00)**					
F3	72 (0.35)	215 (0.12)	**2603 (0.00)**				
F4	520 (0.24)	952 (0.04)	136 (0.15)	**2115 (0.00)**			
C1	52 (0.19)	194 (0.06)	20 (0.22)	185 (0.06)	**1291 (0.00)**		
C2	4 (1.52)	29 (0.09)	59 (0.11)	12 (0.12)	7 (0.22)	**1210 (0.00)**	
C3	195 (0.31)	545 (0.12)	32 (0.28)	486 (0.09)	174 (0.02)	8 (0.19)	**1144 (0.00)**
	F1	F2	F3	F4	C1	C2	C3

*The numbers outside the parentheses are shared molecules between two datasets, and numbers inside are the mean absolute deviation (MAD) of LogS of these shared molecules.

In conventional centralized machine learning application for drug discovery, to include more data, researchers collect data from different sources, and assume the data are independent identically distributed (IID). The IID sampling of the training data is important to ensure that the stochastic gradient is an unbiased estimate of the full gradient. However, the assumption is usually violated due to the high data biases introduced in the measurement of scientific tests, which are conducted by different people in different experimental setting-ups. As shown in Table 1, we collected water solubility datasets from different sources and some of the shared molecules between datasets can have distinct recorded values. For example, dataset F1 and dataset C2 have 4 shared molecules, the values of those molecules in two datasets have a Mean Absolute

Deviation (MAD) of 1.52. The large MAD of shared molecules between datasets may signify the data distribution varies across datasets in some extent. In conventional machine learning, even a model predict perfectly right on one dataset, it can't predict well on another due to the violation of IID. Aside from a shared global model for all datasets, it is viable for federated learning to customize a model for each dataset, which is practically useful to deal with biased/Non-IID data.

Tuning model update frequency

Table 2. Performances of using different federated averaging epochs.

Client /	**Federated averaging epochs**				
Test size	**Every 1**	**Every 5**	**Every 10**	**Every 15**	**Every 20**
F1/611	0.878±0.022	**0.864±0.023**	0.872±0.017	0.888±0.013	0.870±0.012
F2/465	0.543±0.018	**0.517±0.011**	0.522±0.007	0.519±0.009	0.531±0.011
F3/260	0.534±0.007	**0.503±0.011**	0.511±0.002	0.515±0.020	0.530±0.006
F4/212	0.324±0.037	**0.296±0.007**	0.311±0.011	0.304±0.009	0.310±0.008
C1/258	0.299±0.023	**0.277±0.003**	0.280±0.004	0.283±0.004	0.287±0.006
C2/242	0.801±0.024	**0.775±0.004**	0.789±0.007	0.779±0.005	0.786±0.005
C3/229	0.328±0.021	**0.313±0.005**	0.315±0.006	0.319±0.010	0.322±0.004

*The reported performance is the MAE of LogS on test sets in 5 independent runs. Clients F1-4 are the federated training participants, and have 1/10 of the dataset as the test set (8/10 for federated training and 1/10 for validation), while clients C1-3 participate in customization only, and has 1/5 of the dataset as test set (3/5 for customization training and 1/5 for validation). The best performance on each client is highlighted in bold.

A six-layer DNN is constructed as depicted in Figure 2B. Compared with conventional centralized model, federated models has the same network architecture except for that after a given epoch of local training, the model updates of each client will be encrypted and uploaded to the coordinator server, and later the coordinator returns a new model to each client. Therein, the model update frequency, i.e. how many epochs should the client run locally before uploading the encrypted model updates, is an influential hyper-parameter. As shown in Table 2, the model updates in every 5 epochs of local training performances yielded the best predictive performance averaged on 5 independent runs.

Performance comparison

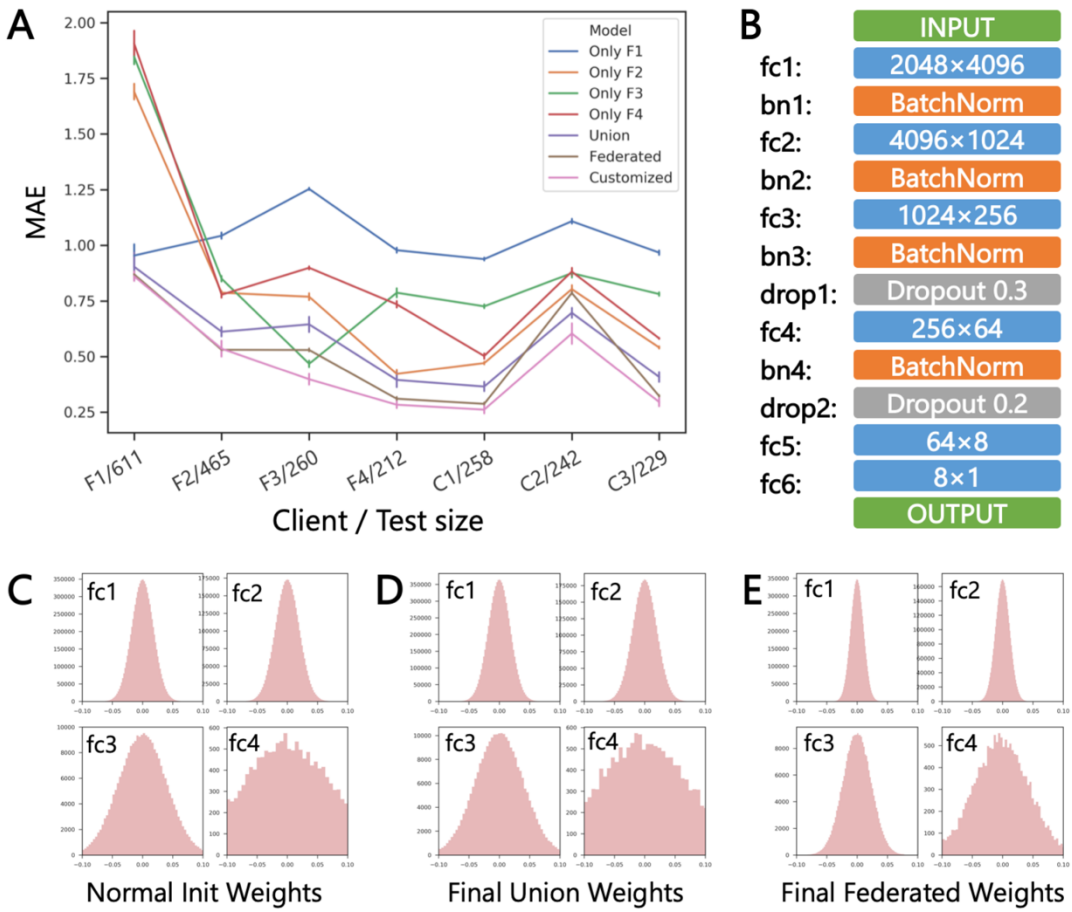

Fig. 2. Performance comparisons between federated modeling and centralized modeling. (A) Centralized models are trained on individual clients (Only F1, F2, F3 and F4) or on the union/pooling of datasets F1/F2/F3/F4 (Union), while federated learning models are trained cross clients F1/F2/F3/F4. The MAE of LogS on the test set of each clients are reported. (B) The deep neural network architecture of the model, which is shared by federated modeling and centralized modeling. Fc is fully connected layer with relu activation. (C) When the weights of fc1-4 are initialized with normal distribution, (D) the weight distributions of union model doesn't vary tangibly after centralized training, while (E) the weight distributions of federated model vary tangibly with more weights concentrated on 0.

In this study, datasets F1-4 were used for simulating clients who participate in the training process of the federated learning models, and C1-3 for simulating clients who didn't participate in training but want to customize the federated model with their own data. We compared federated modeling with individualized and centralized modeling baselines (Figure2 and Supplementary Table 1), in terms of the MAE values on the test set of each client. Generally, the sub-models trained on individual datasets achieved higher performance on their own internal test set (i.e., F1/611, F2/465, F3/260 and F4/212), but much lower performance on other test tests, indicating these sub-models can't generalize well. In contrast, the federated learning model and Union model showed much improved predictive performance on cross client datasets. For clients F1-

4, the federated learning model generally yielded lower MAE values than the corresponding sub-models trained locally, and the prediction capability was maintained on tests from external clients C1-3. It is worth noting here that the federated model performed even better than the Union model, in which data from different sources are simply pooled together for training in a non-privacy-preserving way. This is counter-intuitive, to examine their differences in learning, we compared the weight distribution of fully connected layers in the Union model and the shared Federated model (Figure 2C-E). The weight distributions of the Union model basically unchanged after centralized training compared with the initialized weight distribution, while the weight distributions of federated model vary significantly with more weights concentrated on 0. In the same network architecture with the same cohort of parameters, more weights of 0 means that the model is regularized and simpler, which is likely to generalize better[20,21]. This regularization effect explains the side benefit of federated learning when preserving data privacy on training datasets with different systematic biases. As shown in Table 1, dataset F3 showed a larger systematic bias, where the compounds shared with datasets F1 and F4 have averaged MAD values of 0.35 and 0.24, respectively, which may cause the inferior performance of the Union model and the shared federated model when testing on dataset F3. However, when the shared federated model is further fine-tuned with small learning rate locally without uploading the updates, which is a form of customization for the local data, the performance of the federated model can be further improved, especially for datasets with higher biases (i.e., F3 and C2).

Discussion

In a bigger federated learning context, the framework we set up only focuses on simulating participants who have the same feature space (molecular ECFP fingerprints) as input, which is attributed to horizontal federated learning (Supplementary figure 1A)[22]. There are also vertical federated learning scheme can cope with participants having different feature types as input (Supplementary figure 1B). Moreover, a combination of horizontal and vertical federated learning can effectively handle the participants who share some feature types and samples but also have their own proprietary feature types and samples, which is referred to as federated transfer learning[22]. Federated transfer learning will further expand the feature space and sample size we have by taking both the union of feature space and sample space of multiple participants. For example, to predict the clinical outcome of drug candidates, we need integrate data with shared and proprietary features from multiple parties, including related pharmaceutical companies, hospitals and patients, federated transfer learning may generate large adding value for each party.

Federated learning may still get some security concerns and malicious or non-malicious failures, but it has attracted substantial attention and has been improving and evolving quickly. This paradigm opens up the possibility to integrate confidential datasets through secure distributed training, which have previously been considered

impractical but absolutely attractive to drug discovery. Given the predictive model in drug discovery often work in very confined domain, the opportunity to leverage larger and diverse data silos from multiple institutions will improve generalizability of the predictive models in drug discovery.

Conclusion

In this work, we set up a cross-silo federated learning framework for drug discovery based on FATE[19]. We collected 7 drug solubility datasets and simulated the whole process including federated training, model selection, rollout and customization. Federated training can perform better than individual training on each dataset and, more surprisingly, better than centralized training on the pooled highly biased datasets. Visualizing the weight distributions of parameters in the neural network, we find federated training learned a simpler model with more zero weights than conventional centralized training, which means federated learning intrinsically has a regularization effect and may contribute to better generalization performances in highly biased data. Beyond that, it is feasible for federated learning to customize the global model locally (don't need to upload the model updates) if new users having plenty of labeled data. We demonstrated that users can get some benefits from customizing the global model than using the global model directly. Federated learning represents a new machine learning paradigm, the feature of privacy-preserving will encourage more and more institutions to fully utilize their data and expose more and more data to the latest machine learning models, thus solving the "small data" dilemma in drug discovery. Federated learning setting also makes it feasible to customize models for different users/clients, and hence alleviate the problem of data bias and achieve better predictive performance and form wiser instructions in real application scenario.

Methods

Data curation and partitioning

The 7 aqueous solubility datasets are collected from 7 different sources, which are preprocessed and curated by Sorkun et al. in AqSolDB[23]. Dataset F1 was extracted from eChemPortal, an open-source chemical property database developed by the OECD (Organisation for Economic Co-operation and Development)[24]. Both Dataset F2 and Dataset F4 are obtained EPI Suite Data website, which were generated by Water Solubility Fragment program[25] and WSKOWWIN program[26], respectively. Dataset F3 was taken from the work of Raevsky et al.[27]. Dataset C1 was collected from the work of Huuskonen et al.[28]. Dataset C2 was collected from the work of Wang et al.[29]. Dataset C3 was collected from the work of Delaney et al.[30].

In our simulation, the owners of datasets F1-4 are supposed to be the collaborated parties who want to participate in federated training, and the owners of datasets C1-3 are users who want to use the federated trained model. To prevent overfitting, 1/10 molecules of datasets F1-4 were held out as validation set. Another 1/10 molecules were held out as test sets, for comparing with different models. Because user C1-3 have their own data, it is feasible for them to customize their own model by fine tuning the federated trained model. We set the proportion of train, validation and test sets of E, F, and G by 3:1:1.

Federated Averaging and Secure Aggregation

In our setting, all clients have the same features (molecular fingerprints) as input for prediction task, so all the clients are deployed with the same neural networks architecture and could be trained with Federated Averaging. To ensure the security of data and model, the model updates should also not be uploaded in plaintext. Therefore, a Secure Aggregation protocol are implemented together with Federated Averaging. Both the Federated Averaging and Secure Aggregation protocol are proposed by Google's team in separate works[7,31].

As descripted in pseudo-code of Algorithm 1, when the training start, the coordinator initialize the model parameters w_0, which will be broadcast to each clients. In each round of federated training, the client download the current shared global model w_t from the coordinator server, and trains its model locally on its own data with SGD optimizer. At every E (a hyperparameter) epochs of local training, all the clients will compute the updates and encrypted them under Secure Aggregation protocol. As per the protocol, the local model updates of each client would be added an unique random mask that are carefully generated and relevant to all the other participants, so as to make sure all the random masks adds up to 0 and thus be cancelled out when the coordinator aggregates the local updates uploaded by all clients. Since the random masks are cancelled out, the coordinator gets the true averaged model updates and uses them to update the federated model parameters, obtaining the current shared model. The shared global model will be broadcast to all clients, starting a new round of training. Similar to conventional neural networks, the training process will stop when the federated model converges or the training process reaches a predefined max-round threshold. Note that not as simple as descripted in Algorithm 1, Secure Aggregation Protocol is much more complicated with a four-round interaction between the coordinator and clients, which make protocol robust to dropouts and delays of the clients. The Federated Averaging and Secure Aggregation Protocol are implemented on FATE[19].

Algorithm 1 Federated Averaging with Secure Aggregation

Coordinator Server executes:

initialize w_0

for each round t = 1,2,... **do**

for each client k **in parallel do**

$(\Delta_{t+1}^{k}, \lambda n_k) \leftarrow \text{ClientUpdate}(k, w_t)$

$\Delta_{t+1} \leftarrow \sum_{k=1}^{K} \Delta_{t+1}^{k}$ *// sum of weighted updates*

$\lambda n_{all} \leftarrow \sum_{k=1}^{K} \lambda n_k$ *// sum of weights*

$\Delta_{t+1} \leftarrow \Delta_{t+1} / \lambda n_{all}$ *// averaged updates*

$w_{t+1} \leftarrow w_t + \Delta_{t+1}$

ClientUpdate(k, w): *// executed on client* k

$w_{shared} \leftarrow w$

for each local epoch i from 1 to E **do**

batches ← (data P_k split into batches of size B)

for batch b in batches **do**

$w \leftarrow w - \eta \nabla l(w; b)$

$\Delta \leftarrow \lambda n_k \cdot (w - w_{shared})$ *// weighted updates*

// note Δ is more amenable to compression than w

// then, encrypt it with random variables s

// specified between client k and all other clients u

// by Secure Aggregation protocol

$\Delta \leftarrow \Delta + \sum_{k<u} s_{k,u} - \sum_{k>u} s_{u,k}$

return $\Delta, \lambda n_k$ to server

Neural network architecture and training

As illustrated in Figure 2B, the deep neural network takes the 2048 bit ECFP molecular fingerprints[32] as input and goes through 6 fully connected layers activated with the relu function, between which there are 4 batch normalization layers and two dropout layers with dropout rates of 0.2, 0.3, respectively[20]. The network architecture is shared by all the clients. The federated model is trained with backpropagation and SGD optimizer, where the learning rate starts with 0.01 and follows a 0.68 decay rate[33].

Model selection and customization

The model is selected by looking at the averaged MAE of the held out validation set in the training participants, if it doesn't improve in 30 rounds, the training would stop. When the best shared global federated model is selected, it can be used for prediction directly. However, new users who did not participate in the federated training may customize their own model by getting the licenses and instrumenting the federated training program as done by the participants of federated training. Then, they can fine tune the federated model locally on their own labeled data without uploading the model updates.

AUTHOR INFORMATION

Corresponding Author
*Phone: +86-21-50806600-1303. E-mail: hljiang@simm.ac.cn (Hualiang Jiang)

*Phone: +86-21-50806600-1308. E-mail: myzheng@simm.ac.cn (Mingyue Zheng).

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENT

We gratefully acknowledge financial support from the National Natural Science Foundation of China (81773634 to M.Z), National Science & Technology Major Project "Key New Drug Creation and Manufacturing Program", China (Number: 2018ZX09711002 to H.J.), and "Personalized Medicines—Molecular Signature-based Drug Discovery and Development", Strategic Priority Research Program of the Chinese Academy of Sciences (XDA12050201 and XDA12020368).

Reference

1. Smalley, E. AI-powered drug discovery captures pharma interest. *Nat. Biotechnol.* **35**, 604–605 (2017).
2. Schneider, P. *et al.* Rethinking drug design in the artificial intelligence era. *Nat. Rev. Drug Discov.* (2019) doi:10.1038/s41573-019-0050-3.
3. Chen, B. *et al.* Harnessing big 'omics' data and AI for drug discovery in hepatocellular carcinoma. *Nat. Rev. Gastroenterol. Hepatol.* (2020) doi:10.1038/s41575-019-0240-9.
4. Hunter, A. J., Lee, W. H. & Bountra, C. Open innovation in neuroscience research and drug discovery. *Brain Neurosci. Adv.* **2**, 2398212818799270 (2018).
5. Zhou, Y. Drug development and medical writing in the digital world. *Med. Writ.* **28**, 18–21 (2019).
6. Riley, P. Three pitfalls to avoid in machine learning. *Nature* **572**, 27–29 (2019).
7. McMahan, B., Moore, E., Ramage, D., Hampson, S. & y Arcas, B. A. Communication-Efficient Learning of Deep Networks from Decentralized Data. in *Artificial Intelligence and Statistics* 1273–1282 (2017).
8. Yang, T. *et al.* Applied Federated Learning: Improving Google Keyboard Query Suggestions. *ArXiv E-Prints* arXiv:1812.02903 (2018).

9. Bonawitz, K. *et al.* Towards Federated Learning at Scale: System Design. *ArXiv E-Prints* arXiv:1902.01046 (2019).
10. Kairouz, P. *et al.* Advances and open problems in federated learning. *ArXiv Prepr. ArXiv191204977* (2019).
11. Wang, K. *et al.* Federated Evaluation of On-device Personalization. *ArXiv E-Prints* arXiv:1910.10252 (2019).
12. Jiang, Y., Konečný, J., Rush, K. & Kannan, S. Improving Federated Learning Personalization via Model Agnostic Meta Learning. *ArXiv E-Prints* arXiv:1909.12488 (2019).
13. WeBank. WeBank and Swiss signed cooperation MOU. https://finance.yahoo.com/ news/webank-swiss-signed-cooperation-mou-112300218.html (2019).
14. Li, W. *et al.* Privacy-Preserving Federated Brain Tumour Segmentation. in *Machine Learning in Medical Imaging* (eds. Suk, H.-I., Liu, M., Yan, P. & Lian, C.) 133–141 (Springer International Publishing, 2019).
15. FeatureCloud. FeatureCloud: Our vision. https://featurecloud.eu/about/our-vision/ (2019).
16. Musketeer. Musketeer: About. http://musketeer.eu/project/ (2019).
17. ai.intel. Federated learning for medical imaging. https://www.intel.ai/federated-learning-for-medical-imaging/ (2019).
18. CORDIS, E. Machine learning ledger orchestration for drug discovery. https: //cordis.europa.eu/project/rcn/223634/factsheet/en?WT.mc_id=RSS-Feed& WT.rss_f=project&WT.rss_a=223634&WT.rss_ev=a (2019).
19. WeBank. FATE (Federated AI Technology Enabler). https://github.com/FederatedAI/FATE.
20. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J Mach Learn Res* **15**, 1929 (2014).
21. Smirnov, E. A., Timoshenko, D. M. & Andrianov, S. N. Comparison of Regularization Methods for ImageNet Classification with Deep Convolutional Neural Networks. *2nd AASRI Conf. Comput. Intell. Bioinforma.* **6**, 89–94 (2014).
22. Yang, Q., Liu, Y., Chen, T. & Tong, Y. Federated Machine Learning: Concept and Applications. *ArXiv E-Prints* arXiv:1902.04885 (2019).
23. Sorkun, M. C., Khetan, A. & Er, S. AqSolDB, a curated reference set of aqueous solubility and 2D descriptors for a diverse set of compounds. *Sci. Data* **6**, 143 (2019).
24. OECD. eChemPortal - The Global Portal to Information on Chemical Substances. https://www.echemportal.org/echemportal/propertysearch/addblock_input.action.
25. US EPA. EPI Suite Data. WATERNT (Water Solubility Fragment) Program Methodology & Validation Documents,. http://esc.syrres.com/interkow/Download/WaterFragmentDataFiles.zip.
26. US EPA. EPI Suite Data. WSKOWWIN Program Methodology & Validation Documents. http://esc.syrres.com/interkow/Download/WSKOWWIN_Datasets.zip.
27. Raevsky, O. A., Grigor'ev, V. Yu., Polianczyk, D. E., Raevskaja, O. E. & Dearden, J. C. Calculation of Aqueous Solubility of Crystalline Un-Ionized Organic

Chemicals and Drugs Based on Structural Similarity and Physicochemical Descriptors. *J. Chem. Inf. Model.* **54**, 683–691 (2014).
28. Huuskonen, J. Estimation of aqueous solubility for a diverse set of organic compounds based on molecular topology. *J Chem Inf Comput Sci* **40**, 773 (2000).
29. Wang, J., Hou, T. & Xu, X. Aqueous Solubility Prediction Based on Weighted Atom Type Counts and Solvent Accessible Surface Areas. *J. Chem. Inf. Model.* **49**, 571–581 (2009).
30. Delaney, J. S. ESOL: estimating aqueous solubility directly from molecular structure. *J Chem Inf Comput Sci* **44**, 1000 (2004).
31. Bonawitz, K. *et al.* Practical Secure Aggregation for Privacy-Preserving Machine Learning. in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security* 1175–1191 (Association for Computing Machinery, 2017). doi:10.1145/3133956.3133982.
32. Rogers, D. & Hahn, M. Extended-connectivity fingerprints. *J Chem Inf Model* **50**, 742 (2010).
33. Haddadpour, F., Mahdi Kamani, M., Mahdavi, M. & Cadambe, V. R. Local SGD with Periodic Averaging: Tighter Analysis and Adaptive Synchronization. *ArXiv E-Prints* arXiv:1910.13598 (2019).