Federated Blood Supply Chain Demand Forecasting: A Case Study

Hanzhe Wei
hanzhe.wei@ucalgary.ca
Electrical and Software Engineering
University of Calgary
Calgary, AB, Canada

Na Li
na.li@ucalgary.ca
Community Health Sciences
University of Calgary
Calgary, AB, Canada

Jiajun Wu
jiajun.wu1@ucalgary.ca
Electrical and Software Engineering
University of Calgary
Calgary, AB, Canada

Jiayu Zhou
jiayuz@msu.edu
Computer Science and Engineering
Michigan State University
East Lansing, MI, USA

Steve Drew
steve.drew@ucalgary.ca
Electrical and Software Engineering
University of Calgary
Calgary, AB, Canada

ABSTRACT

Blood transfusion is a commonly used, life-saving medical therapeutics worldwide. A significant challenge is the high variability of supply and demand in blood products, making it difficult to maintain a balance between preventing shortages of blood products and preventing wastage. Recent studies used data-driven methods on demand forecasting for blood products from regional and centralized databases due to regulatory restrictions, which lack the panorama view of the national blood supply and demand picture. Motivated by achieving better policy-making through national blood supply chain demand forecasting, in this paper, we propose to use federated learning (FL) to forecast the demand for platelets through a case study with simulated scenarios considering a national demand and supply network. Our solution facilitates FL with a Long-Short-Term Memory (LSTM) model to make collaborative predictions for future decision-making processes from distributed and regional time-series data. Empirical studies show that FL brings additional performance improvement in various settings, especially for regions with scarcer and shorter data histories. We release the source code for our study at https://github.com/denoslab/fl-blood-supply-chain.

KEYWORDS

federated learning, blood supply chain, time series forecasting

ACM Reference Format:
Hanzhe Wei, Na Li, Jiajun Wu, Jiayu Zhou, and Steve Drew. 2023. Federated Blood Supply Chain Demand Forecasting: A Case Study. In *KDD FL4Data-Mining '23, August 7, 2023, Long Beach, CA, USA.*. ACM, New York, NY, USA, Article 111, 6 pages. https://doi.org/xxxxxxx.xxxxxxx

1 INTRODUCTION

Powered by blood supply chain management (BSCM), blood transfusion saves over 4.5 million lives annually in the US alone [3]. As an essential transfusion medicine, the supply and demand of blood products can be highly variable over time [22]. The supply of blood products is dominantly from donors with planned quantities, while the demand for blood products can vary due to seasonality, holidays, weekend patterns, and so forth [6]. As such, blood suppliers worldwide face significant challenges in maintaining a balance between preventing shortages and wastage of blood products, as well as reducing the overall cost of maintaining an efficient blood supply chain, including routine and urgent deliveries costs, holding costs, and wastage due to the short shelf lives of products such as red blood cells (RBC) and platelets [9]. Addressing these conflicting requirements for orders of blood products requires the blood products inventory to be carefully managed to keep the overall cost low, along with prudently-designed healthcare policies to compensate for the unavoidable costs to ensure the universal accessibility of the blood products.

Data-driven methods provide a promising way to address the aforementioned challenges by forecasting blood products based on historical usage, enabling the calibration of the short- and long-term demands at a finer granularity and leading to leaner while sufficient inventories and reduced operating costs. While several studies have been conducted on demand forecasting for blood products from regional and centralized databases [15], a national understanding of multiple inventories across jurisdictions would leverage their synergies. Therefore, collectively modeling from multiple databases greatly helps in accurate forecasting and policy-making processes, such as subsidizing the costs from a nationwide perspective. However, predictive models generated from state-of-the-art methods are primarily from a regional database covering data serving only one jurisdiction. Healthcare regulatory burdens prohibit a national model from being generated following the traditional methods because health data cannot be shared outside of the jurisdiction by law, mainly due to confidentiality concerns.

Federated learning (FL) offers a decentralized approach to model training, allowing multiple parties to collaboratively train a predictive model without sharing data, therefore keeping data confidentiality. This privacy-aware approach utilizes collective intelligence from distributed datasets and can bring additional benefits in blood supply chain demand forecasting, inventory management, and future policy making. Despite its potential, there has been no study of blood product demand forecasting with FL.

KDD FL4Data-Mining '23, August 7, 2023, Long Beach, CA, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/xxxxxxx.xxxxxxx

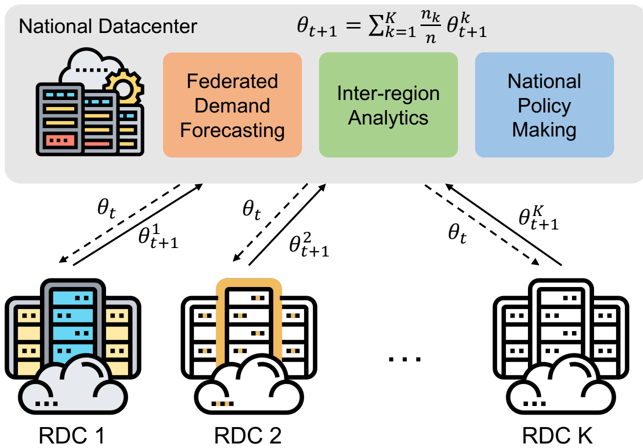

Figure 1: Proposed federated blood supply chain demand forecasting architecture in a case study considering a national blood demand and supply network. At Round t, the National Datacenter transmits the latest global model θ_t and sends it to each regional distribution center (RDC). A local model θ_{t+1}^k is trained at the kth RDC and sent back to the server. The server aggregates all local model updates at Round $t+1$ into a new global model θ_{t+1}. This process repeats until the global model converges or the desired round number is reached.

Inspired by the criticality of blood supply chain demand forecasting and the research gaps identified above, and thanks to the ARIMA models proposed in [15] based on the TRUST database in Hamilton, ON, Canada, in this paper, we explore a federated approach to forecasting the demand for platelets through a case study using simulated data based on the model parameters from [15] without the use of clinical data, for the simplified data accessibility and reproducibility of this study. We identify patterns and trends by considering key factors that affect demands, such as seasonality parameters. We use a Long-Short-Term Memory (LSTM) model that handles sequential information and learns long-term relationships [1], with FL to generate reliable predictions to guide future decision-making processes. Our proposed privacy-aware and collaborative FL approach, shown in Fig. 1, provides the enabling capability for national blood suppliers to manage supply-demand dynamics effectively among local health institutions. To our knowledge, this is the first study that adopts FL for demand forecasting of blood products. Our solution can potentially transform the regulatory landscape of blood chain supply data and greatly benefit the public with such demand forecasting capability. We summarize the following contributions to this study:

- We propose to adopt FL to the unique scenarios of blood supply chain settings where jurisdictional data holders keep their data locally. In a national BSCM network, data remains at the regional distribution center (RDC) level across different provinces. A global model is aggregated nationally based on the local models learned from RDCs nationwide. We believe the shared global model will foster a national view of the demand forecasting, promoting an informed decision-making process for stakeholders.
- We build a time series platelets demand dataset using the ARIMA model with parameters extracted from a large-scale dataset.
- We apply LSTM through FL against the dataset to verify the effectiveness of FL for each local site holding regional data. The experimental results demonstrate the effectiveness of FL in enhancing demand forecasting accuracy, reducing model bias, and improving scalability across heterogeneous datasets and geographically dispersed supply chains.

2 RELATED WORK

Statistical techniques, e.g., autoregressive integrated moving averages (ARIMA), have been extensively used to capture time-series dynamics and are widely accepted for supply chain demand forecasting. However, such traditional methods often rely on a single, centralized source, leading to privacy regulatory concerns [5, 19]. Motamedi et al. [15] highlighted the significance of integrating clinical predictors into demand forecasting models for improving accuracy. Schilling et al. [18] employed a statistical model and a deep neural network to forecast platelet demand within a hospital setting. Their models showed a potential decrease in platelet waste and shortage and substantially reduced financial implications.

Initially proposed by McMahan et al. [14], FL offers a promising solution for training models collaboratively across decentralized data resources. FedProx [10] was proposed to address statistical and system heterogeneity. Due to the wide availability of time series data, FL has been extensively studied with time series data in electrical load forecasting [7, 21] and traffic forecasting [12, 24]. Specifically, the use of LSTM with FL was investigated in [4, 11, 13, 17] for various purposes, including anomaly detection, intrusion detection, and behavior analysis. No study has used LSTM for FL in blood supply chain management.

3 PROPOSED METHOD

In this paper, an ARIMA model is used to generate the data with parameters obtained from [15], as the authors concluded that given suitable parameters, ARIMA models could generate high-quality time series data that reflects the actual blood product demand in a given period with low errors. Other univariate models, e.g., Prophet, and other multivariant models such as Lasso regression, Random Forest, and LSTM were used to forecast the time series data. We recognize that using the ARIMA model for data generation was one approach among several options. We select the ARIMA method in our work due to its effectiveness, interpretability, and wide use in time series analysis.

3.1 The ARIMA model

Effective in practical applications, the AutoRegressive Integrated Moving Average (ARIMA) model is a popular forecasting method for time series data, which captures a suite of different standard temporal structures in time series data, stationary, with a trend, or with a seasonal component. Let $y_1, y_2, \cdots, y_t$ be the demand values over time period t; the time series data can be written as:

$$y_t = f(y_{t-1}, y_{t-2}, y_{t-3}, \cdots, y_{t-n}) + \epsilon_t. \tag{1}$$

An ARIMA model assumes that the value of demand is a linear function of several previous past demand values and previous error values. Thus, the ARIMA model can be written as follows:

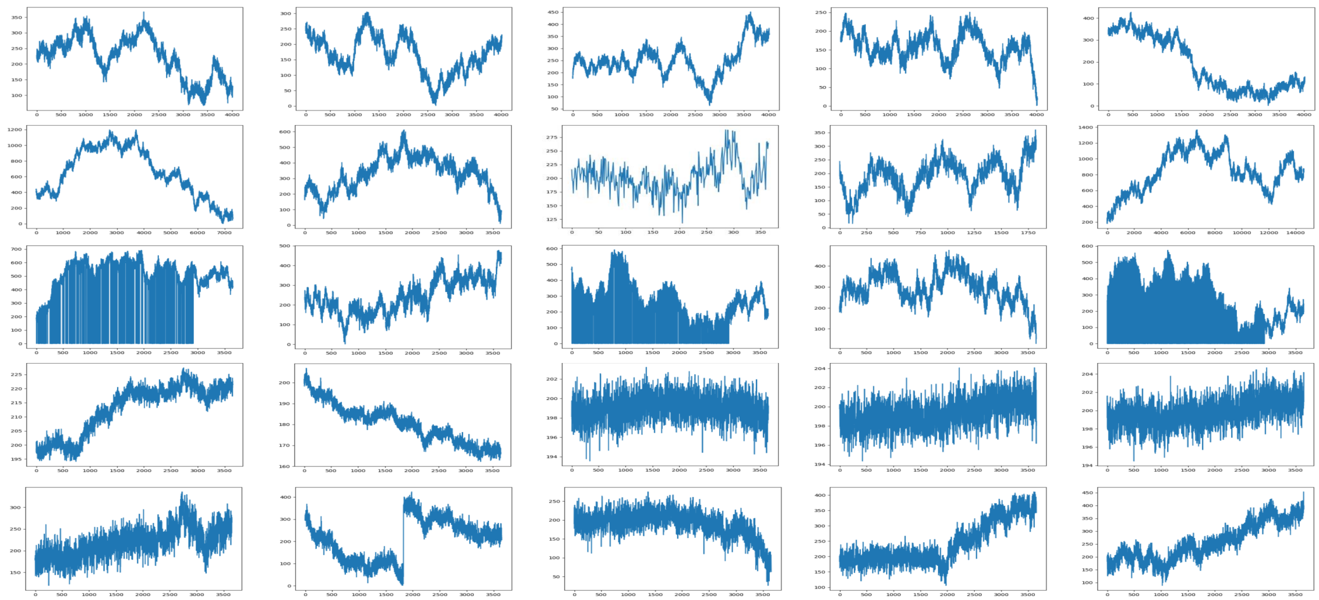

Figure 2: Data generated using ARIMA. First row: balanced i.i.d. data. Second row: imbalanced i.i.d. data. Third row: noisy data with zeros. Fourth row: data with various ARIMA parameters (one set of ARIMA parameters per client). Fifth row: data with various ARIMA parameters (two sets of ARIMA parameters per client concatenated together).

$$\hat{y}_t = \mu + \nu_1 y_{t-1} + \nu_2 y_{t-2} + \cdots + +\nu_p y_{t-p} + \epsilon_t \\ - \phi_1 \epsilon_{t-1} - \phi_2 \epsilon_{t-2} - \cdots - -\phi_q \epsilon_{t-q}, \quad (2)$$

where $\hat{y}_t$ is the response variable (the predicted demand), μ is a constant, ν_i and ϕ_j are model parameters in which $i \in [1, p]$ and $j \in [1, q]$. Here p defines the number of autoregressive terms. and q is the number of lagged forecast errors in the prediction equation. For a nonseasonal ARIMA model, define d as the number of nonseasonal differences needed for stationarity. We can write the model as a function $ARIMA(p, d, q)$ with 3 parameters.

3.2 Synthetic data of platelets demand

In our study, we use the three parameters fitted from the study in [15]. In addition, we generate dataset splits in a heterogeneous fashion under various settings. Fig. 2 shows the time series data generated regarding the demands and time ranges. We list the categories of the synthetic data below with the consideration if the data is independent and identically distributed (i.i.d.):

- **Balanced i.i.d. data**. Each client generates an equal amount of data using the same (p, d, q) values in the ARIMA model.
- **Imbalanced i.i.d. data**. Each client generates different volumes of data with the same parameters. Here we simulate imbalanced data, commonly found in healthcare, as certain institutions may have little or no historical data.
- **Noisy data with zeros**: A portion of clients have random zeros in their datasets, creating noise in the system. This simulates the situation where there are missing or incomplete records.
- **Data with inconsistent ARIMA parameters (non-i.i.d.)**: Data is generated using different ARIMA parameters since institutions may have different demand patterns and unexpected changes in demand patterns in an unprecedented event like COVID-19.

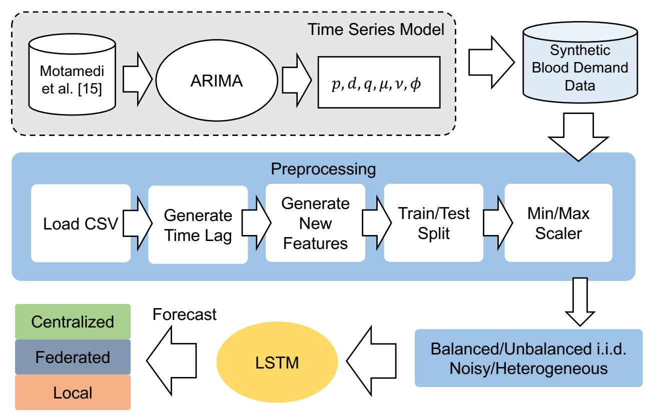

Figure 3: The proposed data pipeline first generates synthetic time series blood product demand data using the ARIMA model learned in [15]. The data is preprocessed before it is split across multiple RDC clients. An LSTM model is then used to learn a federated model.

3.3 The LSTM model

With the synthetic data, We adopt the LSTM model to forecast the demands. Our LSTM model consists of an input layer with 11 - 50 dimensions (previous 7 - 46 days' values, day of month, month of year, day of week, and week of year), three hidden 64×64 layers, with one fully connected layer with an output dimension of 1. For hyperparameter settings, we use 64 for the mini-batch size and 50 for the number of epochs in each round. We use the ADAM optimizer, with the initial learning rate being 0.001. The total number of trainable parameters is 86,337 - 96,321 for the

LSTM model. The hyperparameters used are selected after extensive experiments with different hyperparameter settings (64, 128, and 256) with varied mini-batch sizes to understand how it affects the convergence speed and resource utilization, shown in Fig. 4. As a result, we find that in our case, batch sizes above 64 offer more generalization but not enough precision in their predictions, leading to higher errors; thus, a mini-batch size of 64 is chosen.

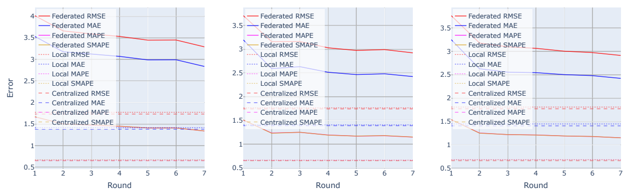

Figure 4: Evaluation results of different batch sizes under heterogeneous settings(Inconsistent ARIMA parameters, two clients share one set while three others share another set). Left: Batch size of 64. Middle: Batch size of 128. Right: Batch size of 256.

Statistical models, such as ARIMA, SARIMA [16], and ARIMAX [23], along with simpler neural network architectures, including the multi-layer perception (MLP) and LSTM [8] models, are widely used in healthcare. Since our data is generated synthetically with an ARIMA model fitted from actual data, we choose the LSTM model as it is commonly used in demand forecasting [20]. Adopting LSTM will lead to generic conclusions showing FL's effectiveness with complex, real-world time-series data.

4 EXPERIMENTAL RESULTS

4.1 Experimental settings

Our experiments use an FL system based on the Flower framework [2]. The experiments are conducted on a Windows PC (AMD Ryzen 7 5800X, 32GM RAM, RTX 4070 with 12GB GDDR RAM), simulating 1 centralized server and 5 local clients. The source code can be found at https://github.com/denoslab/fl-blood-supply-chain.

We apply two popular FL algorithms: FedAvg [14] and FedProx [10]. Fig. 1 shows the overall FL framework, where a client is a regional distribution center (RDC) holding that region's blood supply and demand data. Each client trains its local data based on the global model maintained by the server. We train an LSTM model with the synthetic dataset under the following 5 scenarios in the FL setting. For each scenario, 5 clients are participating in training. Each year of data contains 365 data points.

- **Balanced i.i.d. data scenario (Balanced)**: each client carries 10 years of data generated from the same (p, d, q) values.
- **Imbalanced i.i.d. data scenario (Imbalanced)**: each client holds a different amount of data, carrying 20 years, 10 years, 1 year, 5 years, and 40 years of data, respectively, generated from the same (p, d, q) values.
- **Noisy data scenario with zeros (Noisy)**: 3 out of 5 clients have random zeros in their data, with Client 1 having 20% of data being random zeros, Client 3 having 30% of data being random zeros, and Client 5 having 40% of data being random zeros.
- **Heterogeneous scenario with one set of ARIMA parameters per client (Hetero.)**: two clients share one set of ARIMA parameters while three other clients share another set.
- **Heterogeneous scenario with two sets of ARIMA parameters per client (Hetero. M.)**: the five clients use a series of combinations of ARIMA parameters for their local data.

Algorithm 1 Synthetic data generation with ARIMA. v is the autoregressive term. ϕ is the moving average term. p is the length of v. q is the length of ϕ, μ is the mean of the error term. σ is the standard deviation of the error term. d is the number of unit roots. t is the deterministic linear trend. n is the time series length, and *burn* is the number of discarded values used to start the data generation.

Initialization: Initialize an empty dataframe. Define the AR and MA terms ϕ, v. Define d, t, μ, σ, n, and *burn*.
Error terms: Create an array size of $(n + \max(p, q) + burn)$ with error terms following $\mathcal{N}(\mu, \sigma^2)$.
for i in range(length of error terms array) **do**
  obtain the *AR* term by $\vec{v} \cdot \vec{p_{t-1}}$
  obtain the *MA* term by *current error term* $+ \vec{\phi} \cdot \vec{q_{t-1}}$
  current value is $AR + MA + t$
end for
Unit Root: If $d! = 0$, introduce the unit root to the time series.
Return the last n values of the simulated ARIMA process.

4.2 Numerical Results

With the five data distribution settings above, we compare the following three methods. In the following sections, the terms Federated, Local, and Centralized will be used to refer to these methods.

- **Federated:** A federated model is learned with LSTM using FL.
- **Local:** A model is learned from a single client with LSTM.
- **Centralized:** A centralized model is learned from all clients using a centralized LSTM. This can be considered the ideal learning method that defines the upper bound of learning.

We use the performance metrics of root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and symmetric mean absolute percentage error (SMAPE). Table 1 shows the values of these errors under the two methods, i.e., the Federated method and the Local method, with five data distribution settings, i.e., Balanced, Imbalanced, Noisy, Hetero., and Hetero. M., as defined in Section 4.1.

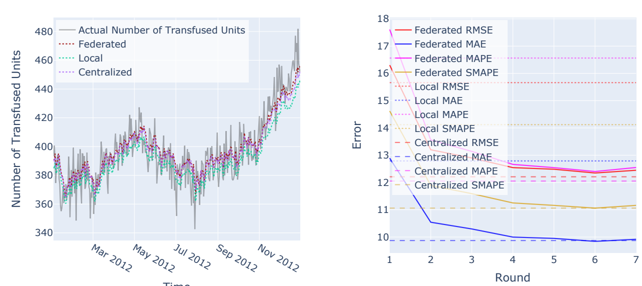

Figure 5: Evaluation results with the balanced i.i.d. dataset. Left: ground truth vs. predicted values. Right: prediction errors.

Table 1: Errors for two methods: Federated and Local, combined with five data distribution settings: Balanced, Imbalanced, Noisy, Hetero., and Hetero. M..

Method	RMSE	MAE	MAPE	SMAPE
Balanced local	15.66	12.79	16.56%	14.12%
Balanced federated	**12.45**	**9.91**	**12.55%**	**11.16%**
Imbalanced local	34.99	28.73	14.63%	13.66%
Imbalanced federated	**20.29**	**15.86**	**7.51%**	**7.50%**
Noisy local	31.58	26.08	21.89%	17.97%
Noisy federated	**27.62**	**22.61**	**19.09%**	**15.88%**
Hetero. local	**2.46**	**2.02**	**1.21%**	**1.20%**
Hetero. federated	6.29	5.97	3.57%	3.50%
Hetero. M. local	22.16	17.47	10.25%	9.86%
Hetero. M. federated	**21.67**	**17.02**	**9.99%**	**9.62%**

Balanced i.i.d. data. When data is balanced across all the clients, the Federated method is expected to perform better than the Local method trained with only local data, as the federated model learns knowledge from participating clients. As illustrated by Fig. 5 and Table 1, the Federated method outperforms the Local method with RMSE by 3.21, MAE by 2.88, MAPE by 4.01%, and SMAPE by 2.96%. The errors of the Federated method converge towards the upper bound with the Centralized method. The results indicate that for RDCs with similar data distributions and comparable data volumes, the Federated method can enhance the global model performance by weighing in more knowledge from nationwide.

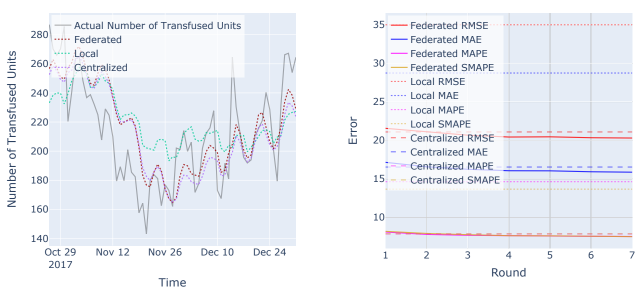

Figure 6: Evaluation results with the imbalanced i.i.d. dataset. Left: ground truth vs. predicted values. Right: prediction errors.

Imbalanced i.i.d. data. When clients have various amounts of data, for those with fewer data samples, the Federated method offers a significant improvement with the global model over clients' local models. This resonates with the case that certain RDCs may have a shorter data history than their counterparts in other regions. Fig. 6 depicts the results of the local and global models on a client with 1 year's worth of data. Compared to the previous setting with balanced data, clients with shorter data histories greatly benefit from FL. At Round 7, the Federated method outperforms the Local method with RMSE by 14.7, MAE by 12.87, MAPE by 7.12%, and SMAPE by 6.16%. For RDCs with imbalanced data, the Federated method proves to be capable of transferring the knowledge from established RDCs to new RDCs.

Noisy dataset with zeros. RDCs may be missing data rows from certain days for various reasons. We explore how the Federated method can improve a noisy dataset with zeroes. When the datasets contain zeros, data imputation preprocessing is done to replace the zero-value row with the first preceding row carrying non-zero values. It is expected to see the Federated method outperforming the Local method under the same condition with the same amount of non-zero values. The evaluation results in Fig. 7 show that the Federated method outperforms Local with RMSE by 3.96, MAE by 3.47, MAPE by 2.8%, and SMAPE by 2.09%. The results confirm that the Federated method can improve demand forecasting accuracy in noisy datasets with missing rows.

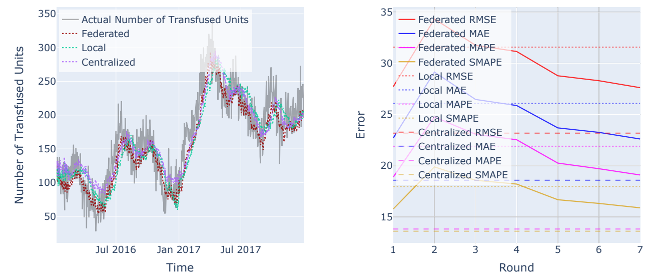

Figure 7: Evaluations with the noisy dataset carrying random zeros rows. Left: ground truth vs. predicted values. Right: prediction errors.

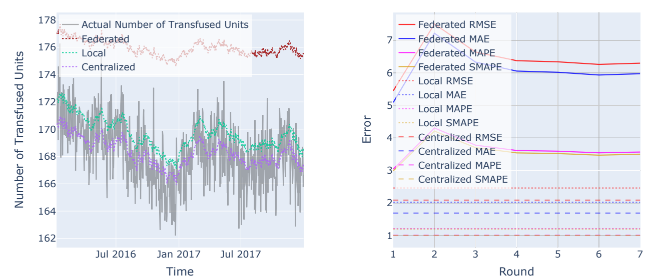

Figure 8: Heterogeneous data with single ARIMA model per client. Left: ground truth vs. predicted values. Right: prediction errors.

Heterogeneous data with single ARIMA model per client. When the ARIMA parameters are different across the clients, given that the data is non-i.i.d. time series data, the Federated method is expected to perform worse under heterogeneous settings. For time series data, if two clients have drastically different trends, the Federated method will give a model that does not do well on either of the clients because federated learning aims for generalization. Given the data heterogeneity, FedProx [10] is used on the five clients, each having a different set of ARIMA parameters. The results, shown in Fig. 8, are just as expected, with the Federated method performing worse than local models.

Combined with the previous experimental setting with zeros, the two cases indicate that when the level of heterogeneity is low and when it is clear where the distributions are different (i.e., different by zero values), data imputation techniques will help improve FL performance. However, when the data is not from the same type of distribution (e.g., ARIMA) or with the same distribution family but different parameters, it is challenging to expect FedAvg to outperform local models. But in these cases, if some data distribution information was known beforehand, there could be an opportunity to improve the performance of FL.

Heterogeneous data with multiple ARIMA models per client. When multiple ARIMA parameters are used inside one client's data, the local model would have difficulty learning those different parameters. We use three sets of ARIMA parameters, denoted by α, β, and γ. From Clients 1 - 5, (α, β), (β, γ), (α, γ), (α, β), (β, γ) are used, respectively. As shown in Fig. 9, the Federated method outperforms the Local method in complex non-i.i.d situations and slightly underperforms the Centralized method. We notice the improvement of the Federated method compared to the previous setting assigning one ARIMA model to each client. The performance change reflects that the Federated method could contribute more significantly with diverse client data.

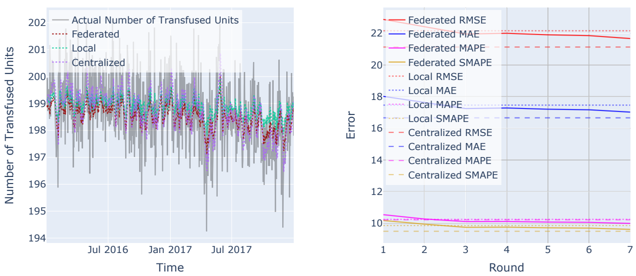

Figure 9: Heterogeneous data with 2 ARIMA models per client. Left: ground truth vs. predicted values. Right: prediction errors.

5 CONCLUSIONS

This paper explored the potential of leveraging FL to enhance blood supply chain demand forecasting. By generating synthetic data with ARIMA models fitted from the real blood supply and demand data, we conducted extensive numerical studies on the benefits of FL with various data distribution settings. When clients have imbalanced i.i.d. data, FL can significantly boost the performance for clients with fewer data points. This is particularly meaningful for newly involved hospitals and RDCs in BSCM. FL improved the overall prediction errors with balanced i.i.d. data, justifying the need for a federated framework. Additionally, FL proved to mitigate the learning accuracy loss caused by missing data. Lastly, FL reduced the errors with heterogeneous data using multiple ARIMA models per client.

ACKNOWLEDGEMENTS

This material is based in part upon work supported by University of Calgary Start-up 10032260, Mitacs Accelerate IT29974, Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant Program (RGPIN-2022-02999), and the NSERC Alliance - Alberta Innovates Advance Program Funding.

REFERENCES

[1] Inci M Baytas, Cao Xiao, Xi Zhang, Fei Wang, Anil K Jain, and Jiayu Zhou. 2017. Patient subtyping via time-aware LSTM networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. 65–74.

[2] Daniel J Beutel, Taner Topal, Akhil Mathur, Xinchi Qiu, Javier Fernandez-Marques, Yan Gao, Lorenzo Sani, Kwing Hei Li, Titouan Parcollet, Pedro Porto Buarque de Gusmão, et al. 2020. Flower: A friendly federated learning research framework. *arXiv preprint arXiv:2007.14390* (2020).

[3] Margaret Carman, Jennifer Schieferle Uhlenbrock, and Sara Marie McClintock. 2018. CE: a review of current practice in transfusion therapy. *AJN The American Journal of Nursing* 118, 5 (2018), 36–44.

[4] Rishu Chhabra, Saravjeet Singh, and Vikas Khullar. 2023. Privacy enabled driver behavior analysis in heterogeneous IoV using federated learning. *Engineering Applications of Artificial Intelligence* 120 (2023), 105881.

[5] Bahareh Fanoodi, Behnam Malmir, and Farzad Firouzi Jahantigh. 2019. Reducing demand uncertainty in the platelet supply chain through artificial neural networks and ARIMA models. *Computers in biology and medicine* 113 (2019), 103415.

[6] Sima M Fortsch and Elena A Khapalova. 2016. Reducing uncertainty in demand for blood. *Operations Research for Health Care* 9 (2016), 16–28.

[7] Nastaran Gholizadeh and Petr Musilek. 2022. Federated learning with hyperparameter-based clustering for electrical load forecasting. *Internet of Things* 17 (2022), 100470.

[8] Shruti Kaushik, Abhinav Choudhury, Pankaj Kumar Sheron, Nataraj Dasgupta, Sayee Natarajan, Larry A Pickett, and Varun Dutt. 2020. AI in healthcare: time-series forecasting using statistical, neural, and ensemble architectures. *Frontiers in big data* 3 (2020), 4.

[9] Na Li, Fei Chiang, Douglas G. Down, and Nancy M. Heddle. 2021. A decision integration strategy for short-term demand forecasting and ordering for red blood cell components. *Operations Research for Health Care* 29 (2021), 100290. https://doi.org/10.1016/j.orhc.2021.100290

[10] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems* 2 (2020), 429–450.

[11] Yi Liu, Sahil Garg, Jiangtian Nie, Yang Zhang, Zehui Xiong, Jiawen Kang, and M Shamim Hossain. 2020. Deep anomaly detection for time-series data in industrial IoT: A communication-efficient on-device federated learning approach. *IEEE Internet of Things Journal* 8, 8 (2020), 6348–6358.

[12] Yi Liu, JQ James, Jiawen Kang, Dusit Niyato, and Shuyu Zhang. 2020. Privacy-preserving traffic flow prediction: A federated learning approach. *IEEE Internet of Things Journal* 7, 8 (2020), 7751–7763.

[13] Yi Liu, Neeraj Kumar, Zehui Xiong, Wei Yang Bryan Lim, Jiawen Kang, and Dusit Niyato. 2020. Communication-efficient federated learning for anomaly detection in industrial internet of things. In *GLOBECOM 2020-2020 IEEE Global Communications Conference*. IEEE, 1–6.

[14] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*. PMLR, 1273–1282.

[15] Maryam Motamedi, Na Li, Douglas G Down, and Nancy M Heddle. 2021. Demand forecasting for platelet usage: from univariate time series to multivariate models. *arXiv preprint arXiv:2101.02305* (2021).

[16] Flávio Fonseca Nobre, Ana Beatriz Soares Monteiro, Paulo Roberto Telles, and G David Williamson. 2001. Dynamic linear model and SARIMA: a comparison of their forecasting performance in epidemiology. *Statistics in medicine* 20, 20 (2001), 3051–3069.

[17] Raed Abdel Sater and A Ben Hamza. 2021. A federated learning approach to anomaly detection in smart buildings. *ACM Transactions on Internet of Things* 2, 4 (2021), 1–23.

[18] Maximilian Schilling, Lennart Rickmann, Gabriele Hutschenreuter, and Cord Spreckelsen. 2022. Reduction of Platelet Outdating and Shortage by Forecasting Demand With Statistical Learning and Deep Neural Networks: Modeling Study. *JMIR Medical Informatics* 10, 2 (2022), e29978.

[19] Mahya Seyedan and Fereshteh Mafakheri. 2020. Predictive big data analytics for supply chain demand forecasting: methods, applications, and research opportunities. *Journal of Big Data* 7, 1 (2020), 1–22.

[20] Mohammad Shokouhifar and Mahtab Ranjbarimesan. 2022. Multivariate time-series blood donation/demand forecasting for resilient supply chain management during COVID-19 pandemic. *Cleaner Logistics and Supply Chain* 5 (2022), 100078.

[21] Afaf Taïk and Soumaya Cherkaoui. 2020. Electrical load forecasting using edge computing and federated learning. In *ICC 2020-2020 IEEE international conference on communications (ICC)*. IEEE, 1–6.

[22] Ana Torrado and Ana Barbosa-Póvoa. 2022. Towards an optimized and sustainable blood supply chain network under uncertainty: a literature review. *Cleaner Logistics and Supply Chain* 3 (2022), 100028.

[23] Kinley Wangdi, Pratap Singhasivanon, Tassanee Silawan, Saranath Lawpoolsri, Nicholas J White, and Jaranit Kaewkungwal. 2010. Development of temporal modelling for forecasting and prediction of malaria infections using time-series and ARIMAX analyses: a case study in endemic districts of Bhutan. *Malaria Journal* 9, 1 (2010), 1–9.

[24] Chenhan Zhang, Shuyu Zhang, JQ James, and Shui Yu. 2021. FASTGNN: A topological information protected federated learning approach for traffic speed forecasting. *IEEE Transactions on Industrial Informatics* 17, 12 (2021), 8464–8474.