Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Supplementary materials for

Yalu WANG, Jie LI, Zhijie HAN, Pu CHENG, Roshan KUMAR, 2025. FedSTGCN: a novel federated spatiotemporal graph learning-based network intrusion detection method for the Internet of Things. *Front Inform Technol Electron Eng*, 26(7): 1164-1179. https://doi.org/10.1631/FITEE.2400932

1 Supplementary experimental results

1.1 Confidence interval and significance testing

From the binary and multi-class classification experiments, it can be observed that the proposed method achieves significant performance in both tasks. To further demonstrate the stability of the method, we conducted 10 trials, using the same dataset for experiments on four different methods. The results were recorded, and confidence intervals were calculated based on these results. Additionally, a t-test was performed to compute the p-value, leading to the formation of Table S1 and Table S2, which present the statistical results for binary classification, and Table S3, which presents the statistical results for multi-class classification.

Table S1 Confidence intervals and *p*-values for binary classification on the NF-BoT-IoT-v2 dataset

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
DeepFed	94.57 ± 2.43 $p < 0.001$	93.43 ± 0.61 $p < 0.001$	92.77 ± 0.80 $p < 0.001$	95.09 ± 1.40 $p < 0.001$
EEFED	92.90 ± 0.98 $p < 0.001$	94.44 ± 1.54 $p < 0.001$	91.05 ± 1.71 $p < 0.001$	91.62 ± 0.76 $p < 0.001$
FedAGRU	93.00 ± 0.71 $p < 0.001$	93.02 ± 0.68 $p < 0.001$	92.59 ± 1.16 $p < 0.001$	91.62 ± 0.51 $p < 0.001$
FedSTGCN	96.77 ± 0.47	96.92 ± 0.39	95.19 ± 0.56	97.23 ± 0.48

Table S2 Confidence intervals and *p*-values for binary classification on the NF-ToN-IoT-v2 dataset

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
DeepFed	92.72 ± 1.76 $p < 0.001$	91.26 ± 0.54 $p < 0.001$	90.18 ± 0.93 $p < 0.001$	91.31 ± 1.19 $p < 0.001$
EEFED	93.15 ± 1.21 $p < 0.001$	92.07 ± 1.11 $p < 0.001$	92.89 ± 1.33 $p < 0.001$	92.47 ± 0.67 $p < 0.001$
FedAGRU	90.27 ± 0.81 $p < 0.001$	87.85 ± 1.49 $p < 0.001$	88.47 ± 0.83 $p < 0.001$	86.82 ± 1.31 $p < 0.001$
FedSTGCN	97.25 ± 0.34	95.04 ± 0.63	97.00 ± 0.40	96.72 ± 0.39

Table S3 Confidence intervals and *p*-values for multiclass classification on two datasets

Method	NF-BoT-IoT-v2		NF-ToN-IoT-v2	
	Weighted recall(%)	Weighted F1-score(%)	Weighted recall(%)	Weighted F1-score(%)
DeepFed	72.20± 0.54 $p < 0.001$	73.57± 2.13 $p < 0.001$	78.13 ± 1.65 $p < 0.001$	74.15 ±2.19 $p < 0.001$
EEFED	80.28 ±1.21 $p < 0.001$	83.01 ±0.89 $p < 0.001$	81.56 ± 0.96 $p < 0.001$	79.68 ± 0.67 $p < 0.001$
FedAGRU	70.02 ± 3.2 $p < 0.001$	68.79 ± 1.92 $p < 0.001$	72.36 ± 3.83 $p < 0.001$	78.12 ± 2.31 $p < 0.001$
FedSTGCN	90.12 ± 0.48	91.78 ± 0.97	91.30 ± 0.76	92.01 ± 0.42

From Tables S1 and S2, it can be observed that FedSTGCN performs the best on both datasets, with accuracy rates of 96.77% and 97.25%, and F1-score of 97.23% and 96.72%, respectively. Its confidence intervals are relatively narrow, indicating stable results. DeepFed and EEFED perform next, with accuracy rates and F1-score ranging from 92.72%–94.57% and 91.31%–95.09%, respectively, but their wider confidence intervals show greater variability in results. FedAGRU performs the worst, with accuracy and F1-score of only 90.27% and 86.82% on the NF-ToN-IoT-v2 dataset, and its wider confidence intervals indicate poorer stability. Overall, the FedSTGCN method performs excellently on both datasets, demonstrating its high accuracy and stability in handling binary classification tasks. While other methods show better performance in certain metrics, their overall performance is not as good as FedSTGCN. Furthermore, the *p*-values for all methods are less than 0.001, indicating that these results are highly statistically significant, further validating the effectiveness of the method proposed in this paper.

From Table S3, it can be observed that on the NF-BoT-IoT-v2 dataset, the FedSTGCN method performs the best, with weighted recall and weighted f-score of 90.12% and 91.78%, respectively, and narrow confidence intervals, indicating stable results. The EEFED method performs next, with weighted recall and weighted F-score of 80.28% and 83.01%. The performance of DeepFed and FedAGRU methods is relatively lower, especially FedAGRU, with weighted recall and weighted F1-score of 70.02% and 68.79%, respectively. On the NF-ToN-IoT-v2 dataset, FedSTGCN also performs the best, with weighted recall and weighted F1-score of 91.30% and 92.01%. EEFED follows, with weighted recall and weighted F1-score of 81.56% and 79.68%. The DeepFed and FedAGRU methods perform relatively poorly, especially FedAGRU, with weighted recall and weighted F1-score of 72.36% and 78.12%, respectively. Furthermore, the *p*-values for all methods are less than 0.001, indicating that the results are highly statistically significant. Overall, FedSTGCN demonstrates higher accuracy and stability in multi-class tasks.

1.2 Computational cost

In addition to the classification performance experiments, this paper also statistics the overhead of various methods to validate the practicality of the proposed method. The overhead includes communication overhead, hardware overhead, and time overhead. Table S4 presents the overhead information for each method on the two datasets. Here, the paper selects 5 clients as the statistical standard for comparing the proposed method with other baseline methods.

Table S4 Various overhead information in the experiments

Method	Training		Communication	Hardware			
	Per epoch (s)	Total time	Communication size (M)	CPU (%)	GPU (%)	Video memory (G)	Memory (G)
DeepFed	21	3 h 23 min	629	52	76	16.08	6.2
EEFED	36	5 h 12 min	472	67	83	18.68	9.9
FedAGRU	29	4 h 2 min	279	49	67	12.76	8.6
FedSTGCN	45	7 h 48 min	954	47	59	21.9	13.1

From Table S4, it can be seen that although FedSTGCN has higher overhead in terms of per epoch, total time, communication size, video memory, and memory compared to the other three methods, it has lower overhead in CPU and GPU usage. The reason for the higher overhead in certain areas for FedSTGCN is that it integrates both LSTM and GCN models, which results in relatively more model parameters in the computation process. Although FedSTGCN has higher overhead in certain aspects, it does not cause an order of magnitude increase in overhead. Moreover, given that its recognition performance is better than the other methods, these additional overheads are completely acceptable.