**Baseline Configurations for Federated Learning Models**

Before delving into the parameter-specific experiments, we established baseline configurations for three distinct Federated Learning (FL) scenarios: Pure FL, Blockchain-secured FL, and Malicious FL. Below is a summary table of the configurations used:

|  |  |  |
| --- | --- | --- |
| **Model Type** | **Command** | **Description** |
| **Pure FL** | **!python main.py -nd 50 -max\_ncomm 100 -ha \*,\*,\* -aio 0 -pow 0 -ko 6 -nm 0 -vh 0.08 -cs 0 -B 10 -mn mnist\_cnn -iid 1 -lr 0.02 -dtx 1 > output\_pure2.txt** | An ideal setup without malicious nodes, using a moderate learning rate to promote steady convergence and a Non-IID dataset distribution for a realistic learning challenge. |
| **Blockchain FL** | **!python main.py -nd 50 -max\_ncomm 100 -ha \*,\*,\* -aio 1 -pow 0 -ko 6 -nm 3 -vh 0.15 -cs 0 -B 10 -mn mnist\_cnn -iid 1 -lr 0.005 -dtx 1 > output\_blockchain2.txt** | A permissioned blockchain environment with a few malicious nodes, employing a higher validator threshold and a lower learning rate for increased stability in the presence of adversaries. |
| **Malicious FL** | **!python main.py -nd 50 -max\_ncomm 100 -ha \*,\*,\* -aio 0 -pow 0 -ko 6 -nm 15 -vh 0.05 -cs 0 -B 10 -mn mnist\_cnn -iid 0 -lr 0.02 -dtx 1 > output\_malicious2.txt** | A challenging environment with a significant number of malicious nodes, using a lower validator threshold and a moderate learning rate, reflecting a scenario with compromised security. |

**Key Parameters:**

* **-nd 50**: Number of participating devices.
* **-max\_ncomm 100**: Maximum number of communication rounds.
* **-ha \*,\*,\***: Dynamic assignment of roles.
* **-aio**: Indicates if all roles are combined (**1**) or separated (**0**).
* **-pow**: Consensus mechanism used (**0** for PoS, **1** for PoW).
* **-ko**: Number of consecutive rounds a device is blacklisted after being flagged.
* **-nm**: Number of malicious nodes in the network.
* **-vh**: Validator threshold for accepting updates.
* **-cs**: Signature checking status.
* **-B**: Batch size for model updates.
* **-mn**: Model name (mnist\_cnn in this case).
* **-iid**: Dataset distribution method (IID or Non-IID).
* **-lr**: Learning rate.
* **-dtx**: Chain resyncing option.

These baseline configurations were selected to mirror typical, ideal, and adversarial conditions in a distributed learning environment. They serve as the foundation from which the effects of parameter variations were investigated to understand the dynamics and resilience of VBFL systems.

**Introduction to Parameter Experiments in VBFL**

**Overview**

In the realm of Federated Learning (FL), particularly in the context of Blockchain-Enhanced Federated Learning (VBFL), fine-tuning various parameters is crucial for optimizing model performance, security, and efficiency. This report delves into a series of systematic experiments designed to assess the impact of different parameters on the learning process and overall model effectiveness. Our investigation spans across three distinct models: Pure Federated Learning (FL), Blockchain-based Federated Learning (Blockchain FL), and Federated Learning with Malicious Nodes (Malicious FL).

**Purpose of Experiments**

The experiments are aimed at gaining deeper insights into how specific parameters influence the behavior and outcome of FL models under various conditions. By altering these parameters, we can explore their roles in learning efficiency, data security, and system resilience against adversarial threats.

**Parameters Under Study**

The parameters analyzed in this research fall into two categories:

1. **Parameters for All 3 Models (Pure, Blockchain, Malicious):**
   * **Data Distribution (-iid):** Examines the impact of Independent and Identically Distributed (IID) versus Non-IID data.
   * **Maximum Number of Communication Rounds (-max\_ncomm):** Assesses how the number of communication rounds influences learning progression and model accuracy.
2. **Parameters Specific to Blockchain Model:**
   * **Validator Threshold (-vh):** Focuses on determining the optimal threshold for validators in the blockchain network.
   * **Consensus Mechanism (-pow):** Compares Proof of Work (PoW) and Proof of Stake (PoS) mechanisms in the context of blockchain efficiency and security.
   * **Chain Resyncing (-dtx):** Analyzes the effect of enabling or disabling chain resyncing on blockchain integrity and model reliability.

**Methodology**

Each parameter is tested through carefully designed simulations, where specific configurations are applied to the three models. The impact of these variations is then meticulously recorded, analyzed, and compared. This approach allows for a comprehensive understanding of each parameter's role and its implications in different federated learning contexts.

**Execution Commands**

For each experiment, specific command lines are used to execute the simulations, varying the parameters accordingly. These commands are crucial for ensuring consistency and reproducibility in our experimental setup.

**Structure of Analysis**

For each parameter, the analysis is structured as follows:

* **Objective:** Outlines the specific goals and intentions behind experimenting with the parameter.
* **Expectations:** States the hypotheses or expected outcomes of the experiments.
* **Experiment Design:** Describes the setup and variations used in the experiments.
* **Commands for Execution:** Lists the exact commands used to run the simulations.
* **Data Analysis:** Provides a detailed examination of the results obtained from the experiments.
* **Comparative Analysis:** Discusses the findings by comparing different settings or models.
* **Implications for Deployment:** Explores how the insights gained from the experiments can be applied in real-world VBFL scenarios.

In conclusion, the comprehensive analysis of these parameters is intended to advance our understanding of VBFL systems. By correlating experimental data with theoretical expectations, this research aims to contribute meaningful insights into the optimization and practical application of VBFL in various domains.

**Parameters for All Federated Learning models**

**1. Data Distribution Impact (-iid)**

**Objective**

The objective of this experiment was to investigate the impact of data distribution on the efficiency and accuracy of Vanilla Blockchain Federated Learning (VBFL) models. The comparison between Independent and Identically Distributed (IID) data and Non-IID data aimed to reveal the inherent challenges and potential advantages that influence learning algorithms within these contexts.

**Expectations**

It was hypothesized that models would struggle more with Non-IID data, reflecting the complexity of real-world data variances. The expectation was that Non-IID conditions would result in lower accuracy or require more iterations for the model to converge, as compared to IID data distribution.

**Experimental Design**

The experiments were conducted by simulating VBFL models under two data distribution conditions:

* IID Condition (-iid 1): Data is homogeneously distributed among nodes, hypothesized to enhance model performance due to uniformity.
* Non-IID Condition (-iid 0): Data is heterogeneously distributed, simulating the imbalance often observed in practical datasets.

**Commands for Execution**

The execution commands varied the -iid parameter while keeping other parameters constant to isolate the effects of data distribution.

|  |  |  |  |
| --- | --- | --- | --- |
| **Variation** | **Code for Pure** | **Code for Blockchain** | **Code for Malicious** |
| IID (1)  &  Non-IID (0) | **!timeout {time\_limit} python main.py -nd 50 -max\_ncomm 100 -ha \*,\*,\* -aio 0 -pow 0 -ko 6 -nm 0 -vh 0.08 -cs 0 -B 10 -mn mnist\_cnn -iid {iid\_value} -lr 0.02 -dtx 1 > {output\_file}** | !timeout {time\_limit} python main.py -nd 50 -max\_ncomm 100 -ha \*,\*,\* -aio 1 -pow 0 -ko 6 -nm 3 -vh 0.15 -cs 0 -B 10 -mn mnist\_cnn -iid {iid\_value} -lr 0.005 -dtx 1 > {output\_file} | !timeout {time\_limit} python main.py -nd 50 -max\_ncomm 100 -ha \*,\*,\* -aio 0 -pow 0 -ko 6 -nm 15 -vh 0.05 -cs 0 -B 10 -mn mnist\_cnn -iid {iid\_value} -lr 0.02 -dtx 1 > {output\_file} |

**Data Analysis (Graphs)**

The graphical analysis reveals the following trends:

* IID Data Analysis: As anticipated, the IID graphs showed a relatively stable accuracy rate for the Pure model. This stability highlights the facilitative nature of uniform data distribution in achieving swift model convergence.
* Non-IID Data Analysis: Contrary to IID, the Non-IID graphs displayed a volatile accuracy pattern across all models, with the Malicious FL model showing the most pronounced fluctuations. This pattern accentuates the challenges posed by heterogeneous data distributions.

**Comparative Analysis**

By contrasting the IID and Non-IID data conditions, several key observations were made:

* Pure Model: Under IID, the Pure model demonstrated expected high accuracy. However, under Non-IID conditions, its performance showed variability, emphasizing the difficulties in learning from diverse data.
* Blockchain Model: Remarkably, the Blockchain model under Non-IID conditions maintained a level of performance resilience, suggesting that the integration of blockchain may provide a buffer against the adverse effects of data imbalance.
* Malicious Model: The Malicious model's performance showed significant disruption under Non-IID conditions, indicating that adversarial actors in the network might exploit the complexities associated with imbalanced data distribution.

**Implications for Deployment**

This experiment has highlighted that while IID conditions offer a simplified lens for model assessment, the true test of a VBFL system's robustness is its performance under Non-IID conditions. The Blockchain model's relative stability in such conditions is promising for real-world applications, suggesting that blockchain could be key to managing imbalanced data distributions. Meanwhile, the vulnerabilities exposed in the Malicious model stress the imperative for robust security measures in VBFL systems to ensure integrity and reliability in the face of real-world data complexities.

**2. Communication Rounds Examination (-max\_ncomm)**

**Objective**

This experiment aimed to explore the influence of the maximum number of communication rounds on the learning curve and the final model accuracy in a federated learning context.

**Expectations**

It was posited that a higher number of communication rounds would likely lead to better convergence and more robust learning outcomes, though possibly at the cost of increased computation time and resources.

**Experimental Design**

We compared the performance of models at two different thresholds of maximum communication rounds—50 and 100—to analyze their impact on the learning process.

**Commands for Execution**

Commands were crafted to initiate the models with varied **max\_ncomm** values, maintaining consistent execution conditions for a controlled comparison.

|  |  |  |  |
| --- | --- | --- | --- |
| **Variation** | **Code for Pure** | **Code for Blockchain** | **Code for Malicious** |
| 50 Rounds  &  150 Rounds | !timeout {time\_limit} python main.py -nd 50 -max\_ncomm **{max\_ncomm\_value}** -ha \*,\*,\* -aio 0 -pow 0 -ko 6 -nm 0 -vh 0.08 -cs 0 -B 10 -mn mnist\_cnn -iid 1 -lr 0.02 -dtx 1 > {output\_file} | !timeout {time\_limit} python main.py -nd 50 -max\_ncomm **{max\_ncomm\_value}** -ha \*,\*,\* -aio 1 -pow 0 -ko 6 -nm 3 -vh 0.15 -cs 0 -B 10 -mn mnist\_cnn -iid 1 -lr 0.005 -dtx 1 > {output\_file} | !timeout {time\_limit} python main.py -nd 50 -max\_ncomm **{max\_ncomm\_value}** -ha \*,\*,\* -aio 0 -pow 0 -ko 6 -nm 15 -vh 0.05 -cs 0 -B 10 -mn mnist\_cnn -iid 1 -lr 0.02 -dtx 1 > {output\_file} |

**Data Analysis**

The evaluation focused on the models' learning trajectories and accuracies as influenced by the variation in communication rounds.

* **50 Rounds Analysis**: Expected to reveal initial learning rates, the models' adaptation to the dataset was observed. The graphs indicated that while models initially adapt and reach certain accuracy levels, there is a notable variance between models, with the Pure model showing faster adaptation compared to the others.
* **100 Rounds Analysis**: With more rounds, the graphs illustrated that models tend to show a gradual increase in accuracy. The Pure and Blockchain models, in particular, displayed a more pronounced improvement, suggesting enhanced learning with additional communication opportunities.

**Comparative Analysis**

The comparative study of 50 versus 100 communication rounds highlighted differences in scalability and efficiency:

* **Pure Model**: As anticipated, the Pure model's accuracy improved steadily with more communication rounds, emphasizing the benefits of extended collaborative learning.
* **Blockchain Model**: Interestingly, the Blockchain model showed a more moderate improvement with additional rounds, pointing towards the potential for blockchain to provide stability even with fewer interactions.
* **Malicious Model**: The Malicious model presented a complex interaction, where additional rounds did not necessarily equate to improved learning, reflecting the nuanced impact of adversarial presence in the learning process.

**Implications for Deployment**

The findings underscore the importance of the **max\_ncomm** parameter in real-world applications. For time-sensitive applications, a lower number of rounds could suffice, particularly if the system can maintain robustness against malicious interference. However, for applications where model accuracy is critical, increasing the number of communication rounds is advisable for achieving a well-converged model.

In conclusion, the **max\_ncomm** parameter is vital in tailoring the performance of federated learning models to specific application requirements, balancing speed, accuracy, and computational efficiency. The insights from this analysis will be instrumental in configuring VBFL systems for diverse application landscapes.

**Parameter for Blockchain Federated Learning model**

**3. Validator Threshold Analysis (-vh)**

**Objective**

This analysis aimed to determine the optimal validator threshold (vh) for balancing security and efficiency within Blockchain Federated Learning (Blockchain FL).

**Expectations**

It was hypothesized that a higher validator threshold would bolster security by selectively validating model updates, potentially leading to greater accuracy at a slower pace. In contrast, a lower threshold was expected to expedite the learning process, albeit with a risk of incorporating lower-quality updates.

**Experimental Setup**

The study investigated two thresholds: a permissive threshold at **vh=0.02** and a restrictive one at **vh=0.1**. These thresholds represent the range from an inclusive to a stringent validation policy.

**Commands for Execution**

Specific commands were used to execute the Blockchain FL model simulations, varying the **-vh** parameter to reflect the desired level of validation strictness.

|  |  |
| --- | --- |
| **Variation** | **Code for Running** |
| 0.02  &  0.1 | **!timeout {time\_limit} python main.py -nd 50 -max\_ncomm 100 -ha \*,\*,\* -aio 1 -pow 0 -ko 6 -nm 3 - vh {vh\_value} -cs 0 -B 10 -mn mnist\_cnn -iid 1 -lr 0.005 -dtx 1 > {output\_file}** |

**Data Analysis**

The analysis focused on the effects of the validator threshold on the accuracy stability of the Blockchain FL model across iterations.

* **Low Threshold (vh=0.02)**: The results showed considerable fluctuations in model accuracy, illustrating the system's sensitivity and inclusive approach. This variability highlighted the potential risks and adaptability of the system to diverse update qualities.
* **High Threshold (vh=0.1)**: The graphs depicted a more stable and consistent accuracy, suggesting that stricter validation effectively filters out suboptimal updates, which could be integral to maintaining the integrity of the learning process.

**Comparative Analysis**

Comparing the two threshold settings revealed key insights:

* **Security vs. Efficiency**: The analysis illuminated the trade-off between the model’s integrity and its agility in learning. The restrictive threshold maintained model stability but potentially at the cost of excluding beneficial updates and slowing convergence.
* **Adaptability to Change**: The comparison also considered how adaptable the Blockchain FL model was to each threshold in a dynamic learning environment, with variations in update quality and network conditions.

**Implications for VBFL Deployment**

The selection of a validator threshold carries profound implications for VBFL deployment, with scenarios dictating the choice:

* **High-Stakes Environments**: For applications where precision and security are paramount, the data suggests favoring a higher threshold, despite a slower learning rate.
* **Agile and Dynamic Environments**: Conversely, environments requiring rapid updates might benefit from a lower threshold, accepting the associated risks for quicker adaptability.
* **Strategic Adjustment**: There's a potential for dynamic adjustment of vh in real-time, responding to network states and performance metrics to strike a balance between swift learning and security.

**Conclusion**

the validator threshold is a pivotal hyperparameter in Blockchain FL. It requires careful calibration to meet the demands of VBFL systems, ensuring resilience against data variability and network threats while optimizing for efficiency. The experimental data underpin the importance of fine-tuning vh to achieve the desired operational balance.

**4. Consensus Mechanism Analysis (Proof of Work vs. Proof of Stake)**

**Objective**

This part of the study evaluates the impact of different consensus mechanisms—Proof of Work (PoW) and Proof of Stake (PoS)—on Blockchain-based Federated Learning (BFL) performance. These mechanisms are crucial for maintaining the integrity and efficiency of the blockchain, thereby directly affecting federated learning outcomes.

**Expectations**

PoW was expected to provide stringent security at the cost of higher computational requirements and slower consensus times due to its intensive nature. PoS was predicted to be more efficient in terms of resource utilization, potentially accelerating consensus formation but possibly at the cost of security.

**Methodology**

The BFL model was tested under two consensus conditions:

* **PoW (pow=1)**: Validators perform computationally intensive tasks to secure the network.
* **PoS (pow=0)**: Validators are selected based on their stake, presumed to require fewer resources.

This section was designed to:

* Assess computational expenses and their impact on model convergence and energy consumption.
* Investigate the level of security each mechanism brings to the model updates' trustworthiness.
* Discuss the scalability and practicality of PoW and PoS in typical VBFL environments.

**Variation and Execution Commands**

Commands varied the **pow** parameter to switch between PoW and PoS, enabling a direct comparison of their influence on BFL.

|  |  |
| --- | --- |
| **Variation** | **Code for Running** |
| (PoS – pow 0,  PoW – pow 1) | **!timeout {time\_limit} python main.py -nd 50 -max\_ncomm 100 -ha \*,\*,\* -aio 1 - pow {pow\_pos} -ko 6 -nm 3 -vh 0.15 -cs 0 -B 10 -mn mnist\_cnn -iid 1 -lr 0.005 -dtx 1 > {output\_file}** |

**Comparative Analysis**

The graphs revealed that:

* **PoW (pow=1)**: The learning process under PoW showed a degree of stability in accuracy, with convergence being achieved consistently across iterations. However, the variability in accuracy suggests a slower adjustment to optimal performance, possibly due to the higher computational demands of PoW.
* **PoS (pow=0)**: PoS demonstrated quicker fluctuations in accuracy, potentially indicating faster consensus but with periods of instability. This could reflect a trade-off between efficiency and the rigorous validation of model updates.

The results indicate that while PoW may contribute to a steadier learning curve, PoS offers the potential for quicker adaptation at the expense of consistency. The PoW mechanism's higher computational cost seems to correlate with a gradual but stable learning process, whereas the PoS mechanism’s efficiency may lead to more dynamic learning behavior with varied accuracy levels.

**Conclusion**

The choice between PoW and PoS consensus mechanisms is pivotal in blockchain-augmented federated learning architectures. It influences not only the model's trustworthiness but also the operational viability of the system. This investigation highlights the need to align the consensus mechanism with the specific objectives of VBFL systems to ensure they are resilient and efficient. The experimental insights will guide developers and researchers in selecting the appropriate consensus mechanism, balancing between the security, speed, and resource efficiency required for effective VBFL deployment.

**5. Chain Resyncing Analysis (-dtx)**

**Objective** This segment evaluates the role of chain resyncing in Blockchain Federated Learning (BFL). Chain resyncing is pivotal for synchronicity across nodes, vital for the accuracy and integrity of a shared learning model.

**Expectation and Hypothesis** It was postulated that suspending chain resyncing (-dtx 1) might induce discrepancies in the ledger, risking model inconsistency and learning inaccuracies. In contrast, active resyncing (-dtx 0) was expected to reconcile differences, maintaining the model's accuracy and blockchain integrity.

**Methodology** A dual-setting experiment was orchestrated:

* With Chain Resyncing (-dtx 0), ensuring active discrepancy resolution.
* Without Chain Resyncing (-dtx 1), eschewing reconciliation measures.

The study aimed to discern the impact of these operational modes on BFL model stability and precision.

**Execution Commands** For the execution, models were run with chain resyncing toggled on and off, with the results directed to output files for subsequent analysis.

|  |  |
| --- | --- |
| **Variation** | **Code for Running** |
| Resync On (0) & Off (1) | **!timeout {time\_limit} python main.py -nd 50 -max\_ncomm 100 -ha \*,\*,\* -aio 1 -pow 0 -ko 6 -nm 3 -vh 0.15 -cs 0 -B 10 -mn mnist\_cnn -iid 1 -lr 0.005 - dtx (dtx\_value) > {output\_file}** |

**Analysis and Comparative Review**

The examination focused on model performance consistency and the learning curve's progression under both resyncing conditions. Results highlighted that:

* Models with resyncing on showed stable and consistent accuracy, validating the hypothesis that resyncing is integral for model integrity.
* Models without resyncing presented more volatility in accuracy, suggesting potential challenges with data integrity and learning effectiveness.

**Results and Implications**

Graphs depicting the experimental outcomes demonstrated the anticipated fluctuations in model accuracy with and without resyncing. The findings prompt a discussion on:

* The potential repercussions of deactivating chain resyncing, particularly the risk of model divergence and data integrity loss.
* The efficiency-security trade-off, considering the faster operation without resyncing against the increased data consistency risks.

**Conclusion**

The study on chain resyncing within BFL highlights the delicate equilibrium between operational efficiency and data fidelity. It offers strategic insights into when it may be advantageous to disable resyncing and when it is essential for the reliability of the learning process. The insights from this analysis are intended to guide the nuanced application or avoidance of chain resyncing in various VBFL network conditions and requirements.