J. Electrical Systems 20-7s (2024): 2516-2523

[1] Mayank Jindal
[2] Madan Mohan Tito Ayyalasomayajula
[3] Dedeepya Sai Gondi
[4] Harish Mashetty

Enhancing Federated Learning Evaluation: Exploring Instance-Level Insights with SQUARES in Image Classification Models

Abstract: - Federated Learning (FL) presents a novel approach within the domain of Machine Learning (ML)—enabling the training of ML models in a distributed manner. This paradigmatic shift ensures the preservation of user privacy by conducting the training process at the edge, where data remains localized on individual devices. This stands in contrast to the conventional centralized paradigm—where users transmit data to a central server for processing. The FL framework operates within a more heterogeneous environment characterized by diverse data distributions across clients, a departure from the engineer-analyzed datasets typical of centralized paradigms. Whereas, evaluation of FL models typically relies on statistical techniques such as accuracy, recall, precision, log loss, and the confusion matrix, alongside visualization methods. While these techniques provide an overview of the model's performance and data utilization—they may lack granularity when comparing models with disparate characteristics. However, the SQUARES technique, offers a more nuanced evaluation of model performance at the instance level. This approach facilitates the examination of data biases, outlier detection, and model behavior during training on individual samples. Therefore, this study presents the development and evaluation of an FL image classification model across various scenarios—utilizing the SQUARES prototype. In addressing these more intricate scenarios, we aim to augment traditional visualizations and metrics, thereby uncovering insights and nuances that may elude standard evaluation methods prevalent in ML benchmarks.

Keywords: Artificial Intelligence, Federated Learning, Machine Learning, Privacy.

1. Introduction

Understanding how Machine Learning (ML) algorithms operate and interpret information is an extensive area of study and development [1]–[3]. The field of Data Visualization (DV) plays a crucial role in providing insights into the processing of these algorithms—particularly when they are perceived as black boxes. In recent years, advancements in Deep Learning (DL) algorithms have significantly enhanced performance across various tasks such as image classification and object detection, often achieving or surpassing human-level accuracy. While many data visualization techniques offer insights into the performance of trained models—they often focus on global aspects of accuracy and training—presenting statistical measures at a broader level, such as through chart lines or confusion matrices commonly found in ML libraries. However, these evaluations lack the capability to analyze individual data samples and their interaction with the model in a comprehensive manner. However, the SQUARES visualization technique, as proposed by [4], addresses this limitation by facilitating the rapid discovery of patterns, biases, and issues within training and test datasets. This approach evaluates a standard centralized ML paradigm, where a single machine processes all data for training and testing purposes. Therefore, in this study, we extend the evaluation of SQUARES to a Federated Learning (FL) scenario—a novel ML paradigm designed to uphold user data privacy by training models at the edge with local data. In our study, we implement a SQUARES prototype to analyze an FL environment generated using the LEAF benchmark [5], utilizing a ten-class filtered dataset derived from the Extended Modified National Institute of Standards and Technology (EMNIST) dataset [6]. Through this analysis, we engage in discussions and conduct quick evaluations facilitated by the SQUARES tool. We focus on enhancing the assessment of models developed through the training process and evaluate the diverse elements' influence within a varied environment during model testing. It enables improved monitoring of both client and server test accuracies, along with providing detailed insights into the performance across different classes and individual samples. This comprehensive approach facilitates the identification of irregular sample patterns, anomalies within test datasets, and enhances the scrutiny of trained models [7]–[12].

The structure of the study unfolds as follows: Section 2 discusses pertinent literature related to our solution, emphasizing data visualization. Section 3 provides background information on crucial areas and technologies essential for comprehending our proposed solution. Section 4 introduces our proposed solution, FED-SQUARES, and outlines the LEAF benchmark utilized for generating FL simulations and models. Section 5 delineates the conducted experiments—the experimental environment, the chosen dataset—FEMNIST, and the evaluation tasks undertaken. Section 6 presents the obtained results and initiates discussions based on these findings. In Section 7,

[1] Independent Researcher, San Jose, CA, USA; mayank.jindal5@gmail.com
[2] Aspen University, Phoenix, Arizona, USA; Email: mail2tito@gmail.com
[3] Artificial Intelligence and Machine Learning, Simplyturn Technologies, Princeton, Texas, USA, sai.gondi@ieee.org
[4] Independent Researcher, Atlanta, GA, USA; Email: harish.mashetty@gmail.com

the conclusions drawn from this study are summarized, accompanied by future directions aimed at addressing existing limitations.

2. Related Works

The sharp rise in interest of ML techniques and recent advancements in the field have sparked numerous endeavors in data visualization research. These initiatives aim to comprehend the inner workings of these models and ascertain the optimal approach for specific tasks through fair measures and effective data visualization techniques such as [13], [14]. Noteworthy contributions include [15]—a versatile solution facilitating comparison among diverse ML models by analyzing various subsets of a shared training and test dataset. This platform provides an extensive array of data visualization options, such as line charts, parallel coordinates, and data histograms, allowing users to select visualizations tailored to their specific analyses. Another significant tool is the [16]—designed to present a deeper view of model performance while facilitating direct inspection of data. [17] organizes examples based on user labels, with color coding indicating the assigned label. [18] arranges test examples above and training examples below, with horizontal placement determined by the model's prediction scores. [19] discusses that users can interact with ModelTracker to gain deeper insights and scrutinize individual examples. Furthermore, [4] offers an unique approach to visualizing and comparing model performance, providing instance-level information and presenting data and model statistics. [20]—developed by the same team as ModelTracker—offers a more straightforward interface with a similar objective. Notably, these prior studies assess their solutions using centralized ML models. However, in our study, we propose to adapt the SQUARES methodology, to evaluate the performance of FL models. This adaptation aims to extend the applicability of visualization techniques to the domain of FL, thus contributing to the advancement of distributed ML paradigms.

3. Background

3.1 Federated Learning

This new approach to ML focuses on preserving data privacy and conducting computations at the edge. Unlike traditional centralized methods, where data is transferred from local devices to a central server for training—in FL—the model is sent to the local devices for training with their respective local data, which remains on the device. After training, the locally trained models are sent back to the server, where they are aggregated using specific functions such as FEDAVG [21]—which combines the model weights to create a new global model. This global model is then distributed to new selected clients. The training process involves iterative rounds or cycles until a stopping criterion is met. In contrast to centralized training, FL deals with heterogeneous data distributions among clients, meaning that each device may not have a balanced amount of data across classes. Additionally, the features in the data may be unrelated to the problem domain, biased, or exhibit stylistic variations based on factors such as origin, user, or region. Consequently, advanced data visualization techniques are crucial for interpreting, debugging, and evaluating FL models.

3.2 LEAF—Federated Learning Benchmark

To conduct our FL simulation, we employed the LEAF benchmark [5], a tool designed specifically for simulating FL model training and evaluation. This benchmark enables the extraction of training information and facilitates testing of ML models in a federated manner. The benchmark comprises various federated datasets organized into client formats, where each client possesses a portion of the overall dataset. During the simulation, all operations are centralized on a single machine. The virtual server dispatches models to virtual clients, each holding a segment of the dataset, and training occurs locally on each client. Furthermore, the LEAF benchmark offers a range of ML models for evaluation purposes. It provides flexibility for structural modifications, allowing researchers to explore new scenarios and techniques. Additionally, the benchmark generates comprehensive data on test/train accuracy and loss metrics between rounds for individual clients and the entire framework. Moreover, it offers insights into processing consumption across clients and datasets. While the benchmark includes visualizations, it lacks detailed instance-level and score-level data, limiting the depth of data inspection during testing and training.

3.3 SQUARES

The SQUARES [4] tool is structured into three primary visualization sections, as depicted in Fig. 1. Positioned at the top, there is a series of horizontal histograms corresponding to each class within the model. Within these histograms, the bars to the left of each class's vertical axis represent the proportion of false negatives, while the slashed bars to the right denote false positives, and the solid-colored bars signify true positives. Vertically, each bar's placement indicates the proportion of samples classified within specific score ranges, aligned with the left axis, portraying the model's prediction scores. Above each class axis, a miniature chart line displays all score curves derived from positive cases, using parallel coordinates. The central segment of the interface incorporates a bar containing information about the evaluated model, encompassing accuracy, precision, recall, false positives/negatives, true positive rates, and the number of classes, alongside dataset statistics such as the number of presented samples. Moreover, the tool facilitates grouping samples within the parallel coordinate chart using stacks/bars (for general grouping), strips (for samples with a certain level of similarity), or squares (offering a more detailed granularity, where each square denotes a sample). At the interface's lower section, a table presents detailed information regarding each tested sample, including image/feature visualization, prediction, ground truth, correctness result, and score prediction for each sample class. Users can select examples within the table, prompting the system to plot corresponding score lines on the parallel coordinate chart for each class inferred by the model. Additionally, the tool allows users to choose between stack/bar, strip, or square modes at the top chart and highlights corresponding examples in the table accordingly, while plotting the corresponding score lines on the chart. In the cited research, a distinct environment was utilized to assess two different ML techniques. The evaluation involved a comparative analysis between the proposed visualization method and the consolidated confusion matrix across various tasks conducted with multiple subjects. The tasks were designed to measure both the speed of analysis and the ease of interpretation regarding the model's responses and comparisons. The findings reveal that the proposed SQUARES method outperforms the conventional approach in experiments involving user interactions.

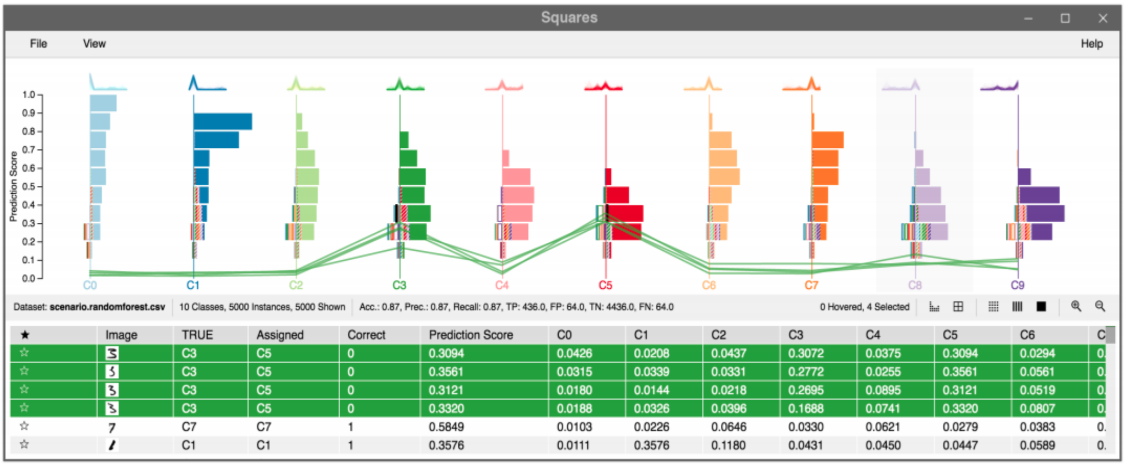

Fig. 1. Original SQUARES interface, showing the results of an FL model analysis for a test sample example

4. Materials and Methods

The FED-SQUARES prototype—as previously indicated—is grounded in the SQUARES visualization, which is a proprietary solution. Our development process for this prototype involved the utilization of various tools. Specifically, we employed D3 for generating charts, Tabulator for constructing tables, and a combination of Javascript, CSS, and HTML. The primary objective of this prototype was to facilitate the comprehension of the model at an individual level. To achieve this, we focused on adapting certain aspects and features of the original tool to suit a FL scenario. These adapted visualizations offer straightforward insights while enabling interactive comparisons between trained models at an instance level. Through these visualizations, users can examine prediction scores for each class, identify mislabelled or corrupted samples, assess model bias, and ascertain the certainty/accuracy levels of the model for each class. Furthermore, users can compare models based on overall statistics and class score inference. In replicating the SQUARES solution for our purposes, we retained key elements such as the top parallel coordinate and horizontal stacked histogram chart. These charts utilize a bar/stacks pattern to depict the proportions of samples within each score bin and their respective class axes. Additionally, essential model information, including the number of classes, accuracy, dataset details, and sample visualizations, is prominently displayed at the top of the interface. The interface design strategically situates the parallel coordinate chart and stacked histogram bars in the middle section, where they illustrate the proportions and scores of false positives/negatives and true positives across all tested samples. At the bottom of the interface, a table presents classification and data details for each sample, retrieved by individual clients and used during testing. Each field within the table is detailed as follows:

1. **Client:** Denotes the source client of the sample data.
2. **Round:** Optionally indicates the number of training rounds completed before sample testing.
3. **Hierarchy:** Represents a parameter concerning client hierarchy, utilized in the simulation tool LEAF (optional).
4. **Num samples:** Refers to the quantity of samples obtained from the client and used for testing.
5. **Set:** Specifies the set name utilized for client evaluation, typically categorized as test, train, or eval.
6. **Features:** Points to the pathway of raw features employed for testing within the simulation, such as an image path.
7. **Accuracy:** Reflects the overall accuracy achieved for the evaluated samples originating from the respective client.
8. **Loss:** Illustrates the overall test loss incurred for the evaluated samples from the specific client.
9. **Ground truth:** Indicates the correct classification label for the particular sample.
10. **Prediction:** Represents the class predicted by the federated model for the specific sample.
11. $\boldsymbol{C_n}$**:** Signifies the prediction score assigned by the federated model to class '*n*' for the specific sample, where '*n*' ranges from 0 to the maximum number of supported classes by the model.

The table enables sorting of rows based on each column header. Additionally, a parallel coordinate chart is provided, showcasing a line with a color corresponding to the predicted class. This visualization aids in evaluating any model confusion in specific examples. Moreover, the image or feature under examination is displayed for inspection in the top left corner of the screen, facilitating visual inspection of the sample and enhancing analysis capabilities.

5. Experimental Analysis

To assess our proposed approach—we conducted experiments utilizing the LEAF benchmark tool—employing a Convolution Neural Network (CNN) integrated into the framework [22]–[28]. The benchmark, while comprehensive, lacked the functionality to retain output layer information alongside test scores. Hence, we modified the benchmark to include this crucial data, ensuring it is saved in a CSV format along with other pertinent information outlined in the previous sections. Additionally, for enhanced compatibility, we converted the CSV file to JSON format, facilitating seamless integration with the table component. The LEAF benchmark tool provides predefined scenarios for experimentation, encompassing factors such as the total number of clients and dataset formats. However, it lacked flexibility in defining the number of classes, prompting us to develop a filter mechanism to adjust class selection and adapt the output layer size accordingly. For our experiments, we selected the FEMNIST dataset, derived from EMNIST [6], tailored for FL evaluations. This dataset comprises 805263 samples distributed among 3,550 users, encompassing 62 classes representing numeric digits and alphabetical characters. To expedite evaluation, we filtered the dataset to include only 10% of clients, each representing 10 classes solely comprising numeric digits. Thus, our filtered dataset comprised 350 clients, and the test set comprised 4056 images, representing 10% of each local dataset from the selected clients. It's noteworthy that dataset distribution among clients was imbalanced, with some clients lacking examples of certain classes. The experiments were conducted on an HP Z840 workstation equipped with an Intel Xeon E5-2609 processor, 32 GB of RAM, and an NVIDIA RTX 2080 GPU. We executed four training runs using the standard LEAF CNN architecture, with modifications to the output layer to accommodate ten classes. The training parameters included a learning rate of 0.001, batch size of 10, and one local epoch for all executions. In the initial three executions, the model underwent training for 500, 1000, and 3000 rounds, respectively. Each round involved selecting two clients with consistent random seeds to ensure reproducibility. This enabled us to monitor the model's evolution, observing class confusion and prediction scores certainty across rounds. Notably, the fourth execution mirrored the second one in terms of the number of rounds (1000), albeit with a different random seed, resulting in varied client selection sequences due to dataset imbalances. We anticipate observing an increase in model accuracy with additional training rounds, facilitating the identification of outliers and mislabeled examples. Furthermore, the fourth execution will enable a comparative analysis of models with similar accuracies, providing insights not attainable through conventional methods such as confusion matrices in centralized environments.

6. Result Analysis

The expected scenario is shown on Fig. 2, where the interface of FED-SQUARES is displayed. The training with 500 rounds achieves 48% of accuracy, presenting in the chart a low level of certainty in the prediction with a maximum score of 0.3 for some samples with a high rate of false positives, represented by the grey bars at the right side of the axis and false negatives, and small colored bars that represent true positives at the right. The results of the three first executions increasing the number of rounds can be seen there. In certain tasks, the model's 41% accuracy can be justified without a direct score visualization; nonetheless, it is much simpler to recognize the poor quality of the model while examining the score prediction. When 1000 rounds are executed, the overall accuracy rises to 79%, the score prediction distribution widens, and responses are almost 100% certain. Similar to the previously mentioned model, this one may satisfy an engineer or customer by merely searching for the confusion matrix or overall accuracy at this level of accuracy. Looking at the ratings, we do observe a significant amount of false positives in class five and false negatives in class eight. Using this model will be risky if these classes are essential to the goal. One could claim that this situation can be examined using a confusion matrix. The FED-SQUARE visualization may point to an issue with the model's score calculation or a high intra-class similarity in this scenario, where the model may produce low scores for each prediction and a high accuracy that is not visible with the confusion matrix. These cases go unnoticed unless the output scores are analyzed. When 3000 rounds are used in the third execution, the model's accuracy rises to 91%. It is evident that there is a low rate of false positives and negatives at this level. The certainty level of every response is also extremely high, with the majority being close to 100%. Since these are common circumstances observed in large datasets labeled by humans, the model can be beneficial, and the incorrect cases can be evaluated to determine if faulty predictions or outliers and miss identified examples are real.

Fig. 2. Visualizations with FED-SQUARES. **Top:** model trained with 500 rounds. **Middle:** model trained with 1000 rounds. **Bottom:** model trained with 3000 rounds

The FED-SQUARES display enables sorting of the data in order to analyze the outliers or mislabeled examples in the prior situation by every table column. Sorting based on the ground truth makes it simple to find incorrect predictions and confirm if an image is distorted, incorrectly labeled, or, in fact, a faulty prediction using the image visualization. As seen in Fig. 3, the left-column example, the line at the chart indicates a high degree of certainty, the digit is readable and displayed at the left corner, and a correct forecast is produced. The right-column example displays an incorrect forecast. The sample with the label 0 was categorized as 1, and upon closer inspection, it is evident that the model is confused by the character's distortion. The digit's thin side raises class 1's score, making it more difficult to determine which class to use. This interface makes it easy to explore and locate examples like this.

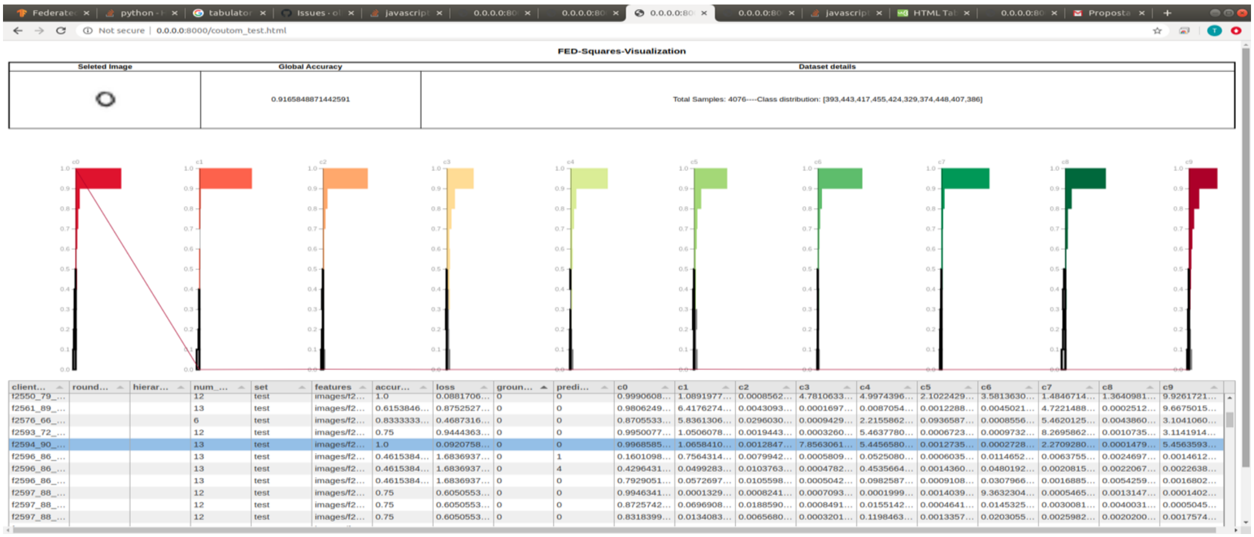

Fig. 3. Inspection of samples predictions with FED-SQUARES

In Fig. 4, two models that were trained with identical parameters but chosen clients in different orders are compared to illustrate the various scenarios of score prediction distribution to each class. Both of them rank in the top two with identical total accuracy (79% and 78%, respectively). It is simple to distinguish between the two models that the stacked parallel chart supports by using FED-SQUARES. It's also simple to see that the second model, with the exception of class 7, shows an overall decrease in score across the board and has less false negatives in class 8 than the first model. The first model has a notable rate of false negatives in class 9. The accuracy models might be somewhat comparable. When compared to a confusion matrix, the models' score prediction distribution is much different. This analysis is also much more visual and understandable, and it is not strictly necessary to investigate every number bin because it is easy to see patterns thanks to the visualization.

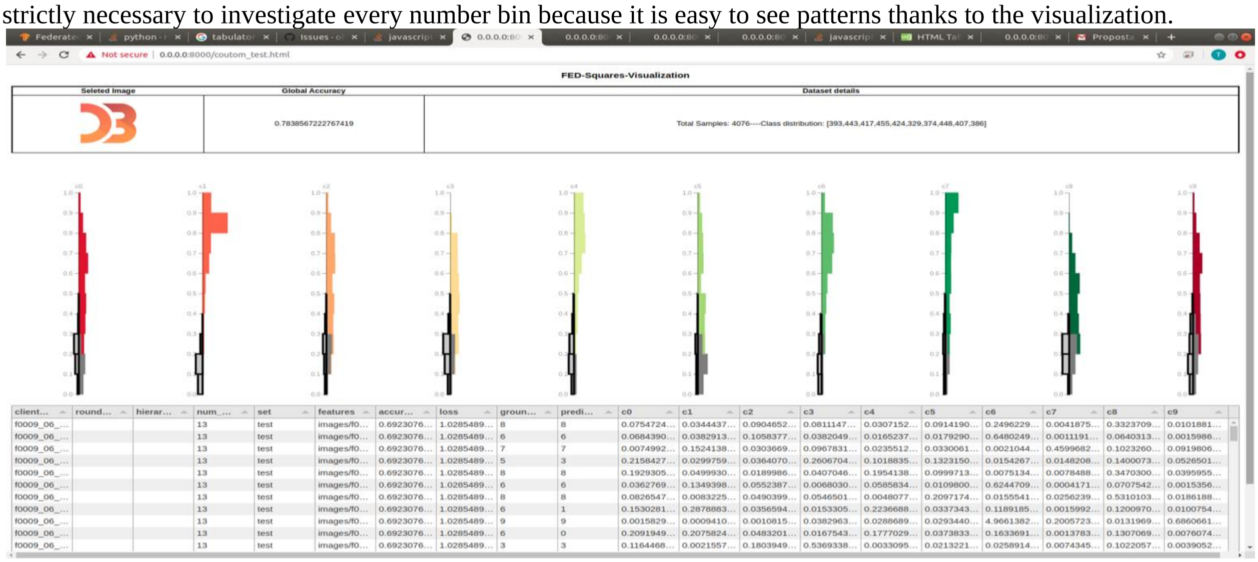

Fig. 4. Models with similar accuracy but very different according to FEDSQUARES

7. Conclusion and Future Works

The foundations of SQUARES, as demonstrated in the original paper, offer a suitable framework for analyzing centered learning scenarios. However, our assessment involves scrutinizing the fundamental concepts and principles of this technique through practical experimentation. We have developed a prototype and conducted simulations within a federated model framework. Our findings align closely with the interpretability of the resultant model, as anticipated. These findings hold promise for enhancing comprehension within FL settings, shedding light on both overall model performance and individual instance-level predictions. Notably, our analyses enable improved monitoring of server and client test accuracy, as well as prediction scores. Interestingly, our observations reveal instances where models with similar accuracies exhibit stark differences in prediction scores and instance-level characteristics. While our evaluation lacks subjectivity, we provide illustrative examples to elucidate our findings. For future endeavors, it would be valuable to incorporate client training dataset insights into our analysis, facilitating a deeper understanding of each client's contribution to the overall model. To improve federated learning frameworks, multi-algorithm strategies and optimization techniques should be the subject of future research. Further advancements [29] – [44] in federated learning models can also be sparked by incorporating sophisticated data segmentation, interactive learning, and reinforcement learning methodologies. Additionally, conducting a subjective evaluation, akin to the approach outlined in the SQUARES, holds significance in definitively validating our proposed solution.

References

[1] N. Kamuni, H. Shah, S. Chintala, N. Kunchakuri, and S. A. O. Dominion, "Enhancing End-to-End Multi-Task Dialogue Systems: A Study on Intrinsic Motivation Reinforcement Learning Algorithms for Improved Training and Adaptability," *2024 IEEE 18th International Conference on Semantic Computing (ICSC)*, pp. 335–340, Feb. 2024, doi: 10.1109/ICSC59802.2024.00063.

[2] N. Kamuni, S. Chintala, N. Kunchakuri, J. S. A. Narasimharaju, and V. Kumar, "Advancing Audio Fingerprinting Accuracy Addressing Background Noise and Distortion Challenges," *2024 IEEE 18th International Conference on Semantic Computing (ICSC)*, pp. 341–345, Feb. 2024, doi: 10.1109/ICSC59802.2024.00064.

[3] N. Kamuni, I. G. A. Cruz, Y. Jaipalreddy, R. Kumar, and V. K. Pandey, "Fuzzy Intrusion Detection Method and Zero-Knowledge Authentication for Internet of Things Networks," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12, no. 16s, pp. 289–296, Feb. 2024, Accessed: Mar. 31, 2024. [Online]. Available: https://ijisae.org/index.php/IJISAE/article/view/4821

[4] D. Ren, S. Amershi, B. Lee, J. Suh, and J. D. Williams, "Squares: Supporting Interactive Performance Analysis for Multiclass Classifiers," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 61–70, Jan. 2017, doi: 10.1109/TVCG.2016.2598828.

[5] S. Caldas *et al.*, "LEAF: A Benchmark for Federated Settings," 2018, Accessed: Apr. 29, 2024. [Online]. Available: http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

[6] G. Cohen, S. Afshar, J. Tapson, and A. Van Schaik, "EMNIST: Extending MNIST to handwritten letters," in *Proceedings of the International Joint Conference on Neural Networks*, Feb. 2017, vol. 2017-May, pp. 2921–2926. doi: 10.1109/IJCNN.2017.7966217.

[7] G. S. Kashyap, A. E. I. Brownlee, O. C. Phukan, K. Malik, and S. Wazir, "Roulette-Wheel Selection-Based PSO Algorithm for Solving the Vehicle Routing Problem with Time Windows," Jun. 2023, Accessed: Jul. 04, 2023. [Online]. Available: https://arxiv.org/abs/2306.02308v1

[8] G. S. Kashyap *et al.*, "Detection of a facemask in real-time using deep learning methods: Prevention of Covid 19," Jan. 2024, Accessed: Feb. 04, 2024. [Online]. Available: https://arxiv.org/abs/2401.15675v1

[9] G. S. Kashyap, A. Siddiqui, R. Siddiqui, K. Malik, S. Wazir, and A. E. I. Brownlee, "Prediction of Suicidal Risk Using Machine Learning Models." Dec. 25, 2021. Accessed: Feb. 04, 2024. [Online]. Available: https://papers.ssrn.com/abstract=4709789

[10] G. S. Kashyap *et al.*, "Revolutionizing Agriculture: A Comprehensive Review of Artificial Intelligence Techniques in Farming," Feb. 2024, doi: 10.21203/RS.3.RS-3984385/V1.

[11] H. Habib, G. S. Kashyap, N. Tabassum, and T. Nafis, "Stock Price Prediction Using Artificial Intelligence Based on LSTM– Deep Learning Model," in *Artificial Intelligence & Blockchain in Cyber Physical Systems: Technologies & Applications*, CRC Press, 2023, pp. 93–99. doi: 10.1201/9781003190301-6.

[12] P. Kaur, G. S. Kashyap, A. Kumar, M. T. Nafis, S. Kumar, and V. Shokeen, "From Text to Transformation: A Comprehensive Review of Large Language Models' Versatility," Feb. 2024, Accessed: Mar. 21, 2024. [Online]. Available: https://arxiv.org/abs/2402.16142v1

[13] H. Wang, J. Ondřejy, and C. O'Sullivanz, "Path Patterns: Analyzing and comparing real and simulated crowds," in *Proceedings - 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, I3D 2016*, Feb. 2016, pp. 49–58. doi: 10.1145/2856400.2856410.

[14] A. Crivellari and A. Ristea, "Crimevec-exploring spatial-temporal based vector representations of urban crime types and crime-related urban regions," *ISPRS International Journal of Geo-Information*, vol. 10, no. 4, p. 210, Apr. 2021, doi: 10.3390/ijgi10040210.

[15] M. Gleicher, A. Barve, X. Yu, and F. Heimerl, "Boxer: Interactive Comparison of Classifier Results," *Computer Graphics Forum*, vol. 39, no. 3, pp. 181–193, Jun. 2020, doi: 10.1111/cgf.13972.

[16] S. Amershi, M. Chickering, S. M. Drucker, B. Lee, P. Simard, and J. Suh, "Modeltracker: Redesigning performance analysis tools for machine learning," in *Conference on Human Factors in Computing Systems - Proceedings*, Apr. 2015, vol. 2015-April, pp. 337–346. doi: 10.1145/2702123.2702509.

[17] A. Arunthavanathan *et al.*, "Support for traceability management of software artefacts using Natural Language Processing," in *2nd International Moratuwa Engineering Research Conference, MERCon 2016*, May 2016, pp. 18–23. doi: 10.1109/MERCon.2016.7480109.

[18] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2004, vol. 3, pp. 2149–2154. doi: 10.1109/iros.2004.1389727.

[19] B. Mwangi, J. C. Soares, and K. M. Hasan, "Visualization and unsupervised predictive clustering of high-dimensional multimodal neuroimaging data," *Journal of Neuroscience Methods*, vol. 236, pp. 19–25, Oct. 2014, doi: 10.1016/j.jneumeth.2014.08.001.

[20] P. Joshi and H. Afli, "Affective Analytics and Visualization for Ensemble Event-Driven Stock Market Forecasting," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2021, vol. 12585 LNCS, pp. 28–55. doi: 10.1007/978-3-030-68007-7_3.

[21] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated Optimization in Heterogeneous Networks," 2018, Accessed: Mar. 17, 2022. [Online]. Available:

https://proceedings.mlsys.org/paper/2020/hash/38af86134b65d0f10fe33d30dd76442e-Abstract.html
[22] S. Wazir, G. S. Kashyap, K. Malik, and A. E. I. Brownlee, "Predicting the Infection Level of COVID-19 Virus Using Normal Distribution-Based Approximation Model and PSO," Springer, Cham, 2023, pp. 75–91. doi: 10.1007/978-3-031-33183-1_5.
[23] G. S. Kashyap, K. Malik, S. Wazir, and R. Khan, "Using Machine Learning to Quantify the Multimedia Risk Due to Fuzzing," *Multimedia Tools and Applications*, vol. 81, no. 25, pp. 36685–36698, Oct. 2022, doi: 10.1007/s11042-021-11558-9.
[24] M. Kanojia, P. Kamani, G. S. Kashyap, S. Naz, S. Wazir, and A. Chauhan, "Alternative Agriculture Land-Use Transformation Pathways by Partial-Equilibrium Agricultural Sector Model: A Mathematical Approach," Aug. 2023, Accessed: Sep. 16, 2023. [Online]. Available: https://arxiv.org/abs/2308.11632v1
[25] G. S. Kashyap, D. Mahajan, O. C. Phukan, A. Kumar, A. E. I. Brownlee, and J. Gao, "From Simulations to Reality: Enhancing Multi-Robot Exploration for Urban Search and Rescue," Nov. 2023, Accessed: Dec. 03, 2023. [Online]. Available: https://arxiv.org/abs/2311.16958v1
[26] N. Marwah, V. K. Singh, G. S. Kashyap, and S. Wazir, "An analysis of the robustness of UAV agriculture field coverage using multi-agent reinforcement learning," *International Journal of Information Technology (Singapore)*, vol. 15, no. 4, pp. 2317–2327, May 2023, doi: 10.1007/s41870-023-01264-0.
[27] S. Wazir, G. S. Kashyap, and P. Saxena, "MLOps: A Review," Aug. 2023, Accessed: Sep. 16, 2023. [Online]. Available: https://arxiv.org/abs/2308.10908v1
[28] S. Naz and G. S. Kashyap, "Enhancing the predictive capability of a mathematical model for pseudomonas aeruginosa through artificial neural networks," *International Journal of Information Technology 2024*, pp. 1–10, Feb. 2024, doi: 10.1007/S41870-023-01721-W.
[29] Kumar, A., Dodda, S., Kamuni, N., & Vuppalapati, V. S. M. (2024). The Emotional Impact of Game Duration: A Framework for Understanding Player Emotions in Extended Gameplay Sessions (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2404.00526
[30] Dodda, S., Kumar, A., Kamuni, N., & Ayyalasomayajula, M. M. T. (2024). Exploring Strategies for Privacy-Preserving Machine Learning in Distributed Environments. Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.36227/techrxiv.171340711.17793838/v1
[31] Kamuni, N., Jindal, M., Soni, A., Mallreddy, S. R., & Macha, S. C. (2024). A Novel Audio Representation for Music Genre Identification in MIR (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2404.01058
[32] Dodda, Suresh, et al. "Exploring AI-Driven Innovations in Image Communication Systems for Enhanced Medical Imaging Applications." Journal of Electrical Systems, vol. 20, no. 3s, 4 Apr. 2024, pp. 949–959, journal.esrgroups.org/jes/article/view/1409/, https://doi.org/10.52783/jes.1409. Accessed 30 Apr. 2024
[33] H. Shah and N. Kamuni, "DesignSystemsJS - Building a Design Systems API for aiding standardization and AI integration," 2023 International Conference on Computing, Networking, Telecommunications & Engineering Sciences Applications (CoNTESA), Zagreb, Croatia, 2023, pp. 83-89, doi: 10.1109/CoNTESA61248.2023.10384889
[34] Dodda, Suresh. "Suresh Dodda: Discussing Automated Payroll Process." Ceoweekly.com, 16 Apr. 2024, ceoweekly.com/discussing-automated-payroll-process-with-suresh-dodda/. Accessed 6 May 2024
[35] Narne, Suman, et al. "AI-Driven Decision Support Systems in Management: Enhancing Strategic Planning and Execution." International Journal on Recent and Innovation Trends in Computing and Communication, vol. 12, no. 1, 16 Mar. 2024, pp. 268–276, www.ijritcc.org/index.php/ijritcc/article/view/10252. Accessed 6 May 2024
[36] Dodda, S., Kumar, A., Kamuni, N., & Ayyalasomayajula, M. M. T. (2024). Exploring Strategies for Privacy-Preserving Machine Learning in Distributed Environments. Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.36227/techrxiv.171340711.17793838/v1
[37] Kamuni, N., A. Cruz, I. G., Jaipalreddy, Y., Kumar, R., & Pandey, V. K. (2024). Fuzzy Intrusion Detection Method and Zero-Knowledge Authentication for Internet of Things Networks. International Journal of Intelligent Systems and Applications in Engineering, 12(16s), 289–296. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/4821
[38] Kumar, A., Dodda, S., Kamuni, N., & Arora, R. K. (2024). Unveiling the Impact of Macroeconomic Policies: A Double Machine Learning Approach to Analyzing Interest Rate Effects on Financial Markets (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2404.07225
[39] N. Kamuni and D. Panwar, "Enhancing Music Genre Classification through Multi-Algorithm Analysis and User-Friendly Visualization," arXiv, 2024, doi: 10.48550/ARXIV.2405.17413.
[40] J. Thomas, "Optimizing Bio-energy Supply Chain to Achieve Alternative Energy Targets," Journal of Electrical Systems, vol. 20, no. 6s. Science Research Society, pp. 2260–2273, Apr. 29, 2024. doi: 10.52783/jes.3176. Available: http://dx.doi.org/10.52783/jes.3176
[41] J. Thomas, "Optimizing Nurse Scheduling: A Supply Chain Approach for Healthcare Institutions," Journal of Electrical Systems, vol. 20, no. 6s. Science Research Society, pp. 2251–2259, Apr. 29, 2024. doi: 10.52783/jes.3175. Available: http://dx.doi.org/10.52783/jes.3175
[42] Suresh Dodda, "Automated Text Recognition and Segmentation for Historic Map Vectorization: A Mask R-CNN and UNet Approach," Journal of Electrical Systems, vol. 20, no. 7s. Science Research Society, pp. 635–649, May 04, 2024. doi: 10.52783/jes.3413. Available: http://dx.doi.org/10.52783/jes.3413
[43] Arpita Soni, "Advancing Household Robotics: Deep Interactive Reinforcement Learning for Efficient Training and Enhanced Performance," Journal of Electrical Systems, vol. 20, no. 3s. Science Research Society, pp. 1349–1355, Apr. 04, 2024. doi: 10.52783/jes.1510. Available: http://dx.doi.org/10.52783/jes.1510
[44] Hemanth Volikatla, "Reinforcement Learning Approaches for Exploring Discrepancy Variations: Implications for Supply Chain Optimization," Journal of Electrical Systems, vol. 20, no. 5s. Science Research Society, pp. 2861–2868, May 03, 2024. doi: 10.52783/jes.3200. Available: http://dx.doi.org/10.52783/jes.3200