Hewlett Packard Enterprise

Hewlett Packard Labs

Check if the document is available in the language of your choice.

Technical white paper

SWARM LEARNING: TURN YOUR DISTRIBUTED DATA INTO COMPETITIVE EDGE

Efficient, secure, privacy-preserving, and decentralized machine learning on blockchain

CONTENTS

EXECUTIVE SUMMARY

The global economy is becoming increasingly digitized with data at its foundation. To unleash the power of data, however, we need to develop algorithms, models, and systems to extract deeply buried insights from data and act upon them.

Machine learning (ML) has enjoyed great success in various applications such as image recognition, object detection, and machine translation. Typical ML architectures use models, trained in public or private clouds using aggregated data to perform inferencing. These models, deployed in the cloud or at the edge, are driving fundamental changes in healthcare, agriculture, retail, transportation, and many other fields. Artificial Intelligence (AI) is driving trillions of dollars of global economic activity,[1] and investment in AI is expected to triple from 2018 to 2022.[2]

In a centralized ML approach, the training data is aggregated to a centralized location, where machine models are developed, trained, and tested. However, the centralized approach is facing mounting technical and socioeconomic challenges. Data sovereignty, security, and privacy can all create barriers to transferring and aggregating the vast amount of data required to train ML models. In addition, the costs of a central infrastructure to host and process the aggregated data can be prohibitive.

The industry is calling for an alternative. The alternative should adapt to, and take advantage of, the increasingly distributed nature of data. It should achieve comparable accuracy to centralized learning but outperform centralized learning in terms of security, fault-tolerance, and latency.

To address this challenge, we have developed a new technology—Swarm Learning.

Swarm Learning is a decentralized ML solution utilizing computing power at or near the distributed data sources with the proven security of the blockchain. In Swarm Learning, both training of the model and inferencing with the trained model occur at edge, where data is most fresh and prompt data-driven decisions are most needed. In its completely decentralized architecture, only learned insights instead of the raw data are shared among collaborating ML peers, which tremendously enhances data security and privacy.

Swarm Learning fundamentally changes the ML computation paradigm by bringing computing close to the data. Its security and privacy-preserving features further open up opportunities for collaboration and monetization models across the organizational boundary.

INTRODUCTION

Data, fueled by ubiquitous sensing, computing, and connection, is driving our economy. Various industries, from healthcare, agriculture, retail, transportation, and many others generate petabytes, if not exabytes of data, every second. But the true value of data comes from the insight, often buried deep, that calls for timely action to create value.

Traditionally, data is aggregated into a central location, typically a public or private cloud, where statistical or ML models are trained using that data. Once trained, the models can be deployed either in the cloud or at the edge, take new inputs, and produce outputs. We call this ML architecture with aggregated training data the centralized ML.

Many machine learning applications, a majority of them utilizing the centralized learning approach, have profoundly affected many aspects of our work and life, bringing us breakthroughs in mobility, lifestyles, and industrial production. As we celebrate the victories of ML applications, we must not lose sight of emerging technical and socioeconomic concerns, and, in particular, their implications on the way that we handle and gain insights from data.

PROBLEM STATEMENT

One major challenge to the centralized learning approach is the increasingly distributed nature of data, driven by the proliferation of data sources around us. An autonomous vehicle, for example, can have LIDAR, radar, vision and many other sensors onboard generating petabytes of data per day. When data is produced at unprecedented speed, in high volume, and at distributed locations, aggregating it to a centralized location, such as cloud, for centralized ML is a formidable if not impractical hurdle. On the other hand, there is a trend toward moving computing, and hence intelligence, closer to data. This edge-first approach could spawn unmatched opportunities in revolutionizing customer experience and establishing competitive advantages, particularly through the shortened delay between data and insight-driven actions. Such opportunities prompt us to explore alternative ML architectures capable of handling highly distributed data and effective in utilizing the rising computing power at the edge.

[1] Visualizing the uses and potential impact of AI and other analytics, McKinsey Global Institute, April 2018
[2] Worldwide Spending on Cognitive and Artificial Intelligence Systems Forecast to Reach $77.6 Billion in 2022, IDC Spending Guide, September 19, 2018

A second hurdle to centralized ML is data privacy and security, which attracts scrutiny from governments, enterprises, and individuals. Consolidating data for centralized ML necessitates moving data, which may get exposed to various attacks during transfer. What's more, aggregation of people's personal information and behavioral habits, captured and dispersed in various data sources such as medical records, browsing history, ride-hailing records, and workout routines will also make it easier to invade a person's privacy. Even when this data is obfuscated or anonymized.[3]

As much as we eagerly seek insights from a vast amount of personal and sensitive data, we need a trusted way to protect individual and enterprise sensitive information. Such a solution should embrace and take advantage of the edge—where data is generated and captured—by empowering the edge to learn right on the spot, without the need to be tethered to a cloud.

Still another limitation of the centralized ML is its data custody model. In many scenarios, a large number of individuals or organizations generate and own the original data, yet the data is collected, cleaned, analyzed, and monetized by another entity. This aggregator has the infrastructure of performing these compute and storage-intensive activities. Most of the time, the aggregator will also serve as the curator. The separation of data ownership from data access and usage curation creates data monopolies, which favors the data aggregators pocketing a large share of data values.

Furthermore, when data owners cede their control of the data-to-data aggregators, the Pandora's Box of privacy invasion is open, despite data use and privacy agreements between the owners and the aggregators. The recent Facebook-Cambridge Analytica data scandal is just one example.

Because of the inherent flaw in the data custody model, data collaboration across organizational boundaries has been especially challenging with the centralized ML approach. Organizations tend to have data centered on a particular industry, customer base, or geography, which, when pooled together, could uncover significantly deeper insights for all of them. Credit card fraud, for example, was estimated to cost financial institutes $33.7 billion worldwide in 2017.[4]

Current ML models for risk detections are developed by each bank using their proprietary data and still suffer from high percentage of false positives. A recent study reported that roughly 1 in 15 (6.7%) cardholders were affected by fraud-related false positives in 2017.[5] In training fraud detection models, the number of fraud transactions is typically limited, which can lead to limited model accuracy. There is, therefore, a great potential for financial institutes to share card transaction data and to improve fraud detection accuracy.

A decentralized system architecture capable of overcoming these challenges would have far-reaching industry influence. The significance of such a decentralized solution is demonstrated in part by the heavy investment in and impact of AI across the global economy. McKinsey, for example, predicts that deep learning alone will have $3.4 trillion to $5.7 trillion total impact across industries by 2020.[6] IDC, on the other hand, projects that spending on cognitive and AI systems will more than triple its 2018 estimate of $24 billion to $77.6 billion in 2022.[7] This offers solid financial returns. According to Deloitte, 82% of the enterprises they surveyed have gained a financial return from their AI investments.[8]

SUCCESS CRITERIA FOR THE SOLUTION

The drawbacks of centralized ML stem from its centralized architecture. These deficiencies naturally prompt us to explore its opposite—the decentralized ML approach. In particular, we need a decentralized machine learning approach with the following attributes:

Effective

Decentralized ML should be effective when gauged on accuracy, efficiency, and capability of handling distributed data.

Accuracy: It should achieve comparable, if not the same, model accuracy compared to centralized learning since accuracy is the main indicator of how well knowledge embedded in the data is captured.

Efficiency: It is critical to look at efficiency from a systematic and end-to-end perspective. An ML system could span data/model parameters transfer, model training/test, deployment, and periodic updates. A fair comparison should also go beyond temporal efficiency to include efficient use of existing compute, storage, and communication infrastructure.

[3] Re-identification Risks in HIPAA Safe Harbor Data: A study of data from one environmental health study, (Sweeney L, Yoo JS, Perovich L, Boronow KE, Brown P, Green Brody J.) Technology Science, August 28, 2017
[4] The Nilson Report, November 2018 (Issue 1142)
[5] Overcoming False Positive Declines in E-Commerce, Javelin LLC, December 2018
[6] Visualizing the uses and potential impact of AI and other analytics, McKinsey Global Institute, April 2018
[7] Worldwide Spending on Cognitive and Artificial Intelligence Systems Forecast to Reach $77.6 Billion in 2022, IDC Spending Guide, September 19, 2018
[8] State of AI in the Enterprise, Deloitte (Second Edition), 2018

Handling distributed data: Intrinsically, decentralized ML needs to be effective in handling distributed data by allocating workloads, coordinating peers, and synthesizing partial learning results to generate the complete model. In addition, the algorithm itself should work with biased or unbalanced data, both of which can be very challenging as they have a direct impact on model convergence during model training and retraining. It is not uncommon that data with certain attributes are not evenly distributed across all data sources. Besides, we should also expect that data volume could vary considerably among distributed data sources.

Secure

Decentralized machine learning needs security features to ensure that only authenticated participants can join the learning, thus protecting distributed data as well as the model (hyper) parameters/weights. To mitigate rouge participants, there should be a mechanism to remove malicious actors as needed and to ensure that data access is limited to authorized duration and purpose only. What's more, additional measures might be necessary to hide the details of the ML model.

Privacy-Preserving

One fundamental motivation for decentralized learning is to provide better privacy. Successful decentralized ML should give data owners better control over access to their sensitive information and extract insight from their data without invading their privacy.

Fault-Tolerant

A centralized ML approach has the risk of a single point of failure. Decentralized learning, although not subject to this risk, needs features to enhance its robustness and to handle the potential dynamic joining and leaving of distributed data resources during model training.

HOW DO EXISTING SOLUTIONS FALL SHORT?

ML has made tremendous progress in both industrial application and academic research over the last decade. There have been efforts toward tackling individual challenges facing the centralized ML. However, a solution that tackles all the major challenges of centralized machine learning has yet to emerge.

Both Google™'s federated learning[9] and Facebook's elastic averaging SGD[10] have explored ways in which local learning can collaborate to improve a shared model. In the work from both Google and Facebook, a single parameter server takes the responsibility of aggregating and distributing local learning. This star-shaped system architecture creates an obvious single point of failure, which leads to reduced fault-tolerance.

Apart from this drawback, a deeper concern for any decentralized ML involving a large number of peers is how to identify, penalize, and exclude the rogue participants. While technical measures help detect rogue participants and limit their influence, the detection and system recovery take time. To deter such malicious behaviors in the first place, we need to look beyond technical means to ensure that each local learning peer does not have the motivation to misbehave as it will incur undesirable consequences.

On the privacy front, the work from UT Austin/Cornell[11] and CMU/Mitsubishi,[12] for example, brings privacy consideration into ML in multiparty settings. This is a major step toward addressing the privacy requirement in decentralized ML. Since the work is understandably focused on privacy improvement, critical issues—such as how to handle the most challenging pattern of distributed data including unbalanced data volumes or non-independent and identically distributed (IID) data that enterprises may encounter—were left out. To eliminate the potential burden of reshuffling or reorganizing distributed data, we need a decentralized ML approach natively developed for distributed data that any privacy improvement work can rest upon.

We also find a gap in the capabilities of existing decentralized ML frameworks for enterprise applications. Google's federated learning, for example, targets an individual end-user's data in its ecosystem. Decentralized ML,[13] a recent startup, also has a consumer-centric focus by crowdsourcing decentralized ML. There is great potential for a complete decentralized machine learning solution for enterprises, where the compute, storage, and data coexist to realize its full potential. This solution should not only address the particular challenges specific to enterprises but also facilitate novel business models to encourage collaborations across enterprise boundaries.

[9] Communication-Efficient Learning of Deep Networks from Decentralized Data, (McMahan B, Moore E, Ramage D, Hampson S, and Blaise Agüera y Arcas), 2017

[10] Elastic Averaging SGD in Distributed Deep Learning, (Sixin Zhang, Anna Choromanska, and Yann LeCun), NIPS 2015

[11] Privacy-Preserving Deep Learning, (Reza Shokri, Vitaly Shmatikov), proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, 2015

[12] Privacy preserving probabilistic inference with Hidden Markov Models, (Manas Pathak, Shantanu Rane, Wei Sun, and Bhiksha Raj), Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and Signal Processing

[13] decentralizedml.com

HPE'S ANSWER TO THE CHALLENGE—SWARM LEARNING

Drawing its inspiration from biology, Swarm Learning is a decentralized ML solution built on blockchain technology, particularly designed to enable enterprises to harness the power of distributed data while protecting data privacy and security. Swarm Learning leverages the computing power at or near the distributed data sources, ensures security using tested blockchain technology, and protects privacy by sharing insights captured from data instead of the raw data itself.

Swarm Learning's blockchain-based security framework ensures that only legitimate participants join the decentralized learning network and that each party is bound by a smart contract in terms of contribution and rewards. The smart contract in Swarm Learning supports innovative business models; along with a monetization framework, it also facilitates cross-organization collaboration.

INTRODUCING SWARM LEARNING TECHNOLOGY

Swarm Learning is a framework designed to make it possible for a set of nodes—each node possessing some training data locally—to train a common ML model collaboratively **without sharing the training data itself**. This can be achieved by individual nodes sharing parameters (weights) derived from training the model on the local data instead. This allows nodes to maintain the privacy of their raw data.

Parameters shared from all the nodes are merged to obtain a global model. Moreover, the merge process is not done by a static central coordinator or parameter server, rather a temporary leader chosen dynamically among the nodes is used to perform the merge, thereby making the Swarm network decentralized. This provides a far greater fault-tolerance than traditional centralized-parameter-server-based frameworks. With the global model, the nodes have the collective intelligence of the network at their disposal, without the data ever leaving the node.

Swarm Learning builds on top of two proven technologies—distributed ML and blockchain. Distributed ML is leveraged to train a common model across multiple nodes with a subset of the data located at each node—commonly known as the data parallel paradigm in ML—though without a central parameter server. Blockchain lends the decentralized control, scalability, and fault-tolerance aspects to the system to enable the framework to work beyond the confines of a single enterprise, and at the same time, introduces a tamperproof cryptocurrency framework, which the participating entities can utilize to monetize their contributions.

SWARM LEARNING WORKFLOW

The Swarm Learning workflow can be divided into three major operational phases:

1. Initialization and onboarding
2. Installation and configuration
3. Integration and training

Each of these operations is described here:

Initialization and onboarding

Onboarding is an offline process that involves multiple entities interested in a Swarm-based ML to come together and formulate the operational and legal requirements of the decentralized system. This includes aspects such as data (parameter) sharing agreements, arrangements to ensure node visibility across organizational boundaries of the entities, and a consensus on the expected outcomes from the model training process. Values of configurable parameters provided by Swarm, such as the peer-discovery nodes supplied during boot up and the synchronization frequency among nodes, are also finalized at this stage. Finally, the common model to be trained and the reward system (if applicable) should be agreed upon.

Installation and configuration

Once the process of onboarding finishes, all the consortium members download and install the Swarm platform on their respective machines (nodes), during which the configuration of the Swarm Learning network finalized during initialization and onboarding step is also supplied. Afterward, the Swarm Learning platform boots up and initiates the node's connection to the Swarm network, which is essentially a blockchain overlay on the underlying network connection between the nodes. The boot-up is an ordered process in which the set of participant nodes designated as peer-discovery nodes (during the initialization phase) are booted up first, followed by the rest of the nodes in the network.

Integration and training

Swarm Learning provides a set of simple APIs to enable swift integration with multiple frameworks. These APIs are incorporated into the existing code base to quickly transform a stand-alone ML node into a Swarm Learning participant. The process of model training can be divided into the following (Figure 1):

1. Enrollment

The Swarm Learning process begins with enrollment, or registration, in the Swarm smart contract by each node. This is a one-time process. Each node subsequently records its relevant attributes in the contract such as the uniform resource identifier (URI) from which its own set of trained parameters can be downloaded by other nodes.

2. Local model training

Nodes next proceed to train the local copy of the model iteratively over multiple rounds, each such round being called an epoch. During each epoch, every node trains its local model using one or more data batches for a fixed number of iterations. After the number is reached, it exports the parameter values in a file and uploads it to a shared file system for other nodes to access. Subsequently, it signals other nodes that it is ready for the parameter-sharing step.

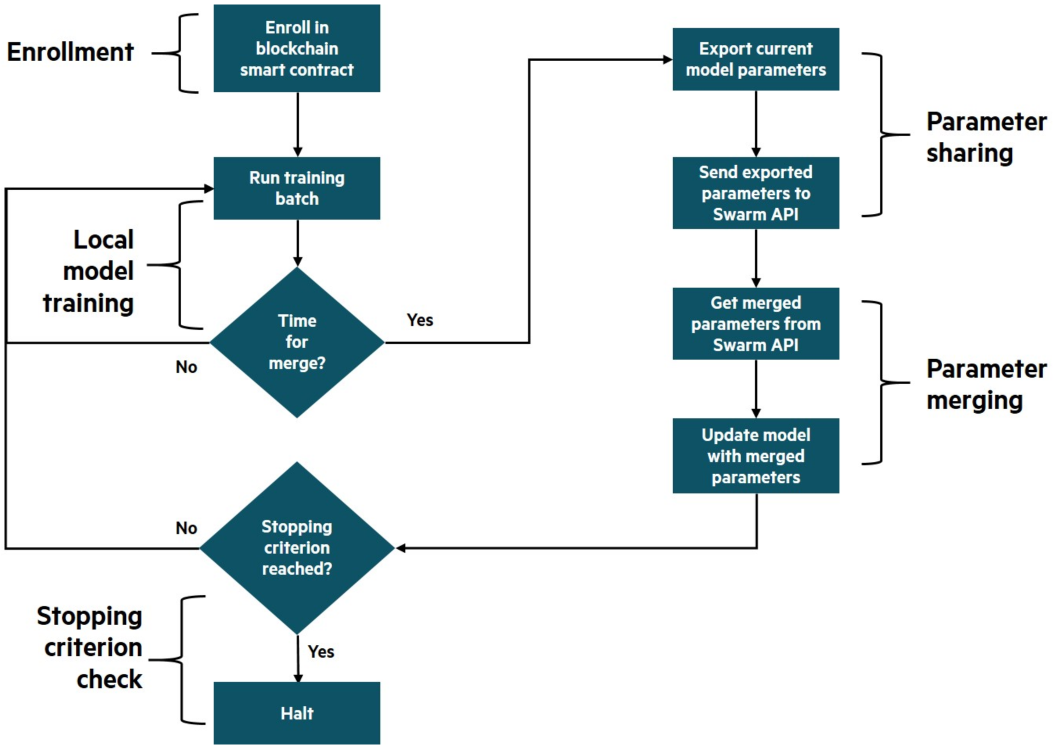
Enrollment
Enroll in blockchain smart contract
Run training batch
Local model training
Time for merge?
No
Yes
Export current model parameters
Send exported parameters to Swarm API
Parameter sharing
Get merged parameters from Swarm API
Update model with merged parameters
Parameter merging
Stopping criterion reached?
No
Yes
Stopping criterion check
Halt

FIGURE 1. Model training steps

3. Parameter sharing

This step commences once the number of nodes that are ready for parameter sharing step reaches a certain minimum threshold value specified during initialization. It begins with the process of electing the **epoch leader**, whose role is to merge the parameters derived after local training on all nodes. This selection is extremely quick and takes place at the culmination of each epoch.

Using the predetermined leader election algorithm, one of the nodes emerges as a leader and then uses the URI information of all the participants, downloads the parameter files from each of them to enable the parameter-merging step. We use a star topology, where a single leader performs the merge; other topologies such as a k-way merge where the merge is carried out by a set of nodes are also possible and easily configurable.

4. Parameter merging

The leader then merges the parameter files downloaded. The framework supports multiple merge algorithms such as mean, weighted mean, median, and so on. Using the merge algorithm chosen, the leader combines the parameter values from all nodes to create a new file with the merged parameters and signals to the other nodes that a new file is available. Each node then downloads the file from the leader and updates its local model with the new set of parameter values.

5. Stopping criterion check

Finally, the nodes evaluate the model with updated parameter values using their local data to calculate various validation metrics. The values obtained from this step are shared using the smart contract state variable. As each node completes this step, it signals to the network that the update and validation step is complete. In the interim, the leader keeps checking for the update complete signal from each node. When it discovers that all merge participants have signaled completion, the leader merges the local validation metric numbers to calculate the global metric numbers. The synchronization step is then marked as complete.

If the policy decided during initialization supports monetization during model building, the rewards corresponding to the contributions by each of the participants are calculated and dispensed at this point. Afterward, the current state of the system is compared against the stopping criterion and if it is found to be met, the Swarm Learning process is halted. Otherwise, the steps of local model training, parameter sharing, parameter merging, and stopping criterion check are repeated until the criterion is fulfilled.

SWARM LEARNING ARCHITECTURE

The architecture of Swarm Learning can be divided into the following four layers—API, control, data, and monetization. The components are modular so that the technologies used in implementing them can be replaced based on requirements. The entire framework is designed to run on both commodity and high-end machines, supporting a heterogeneous set of infrastructure in the network. It can be deployed within and across data centers, and has built-in support for a fault-tolerant network, where nodes can exit and reenter the Swarm network dynamically without derailing or stalling the model building process.

Figure 2 displays the stacking of various Swarm Learning layers, each of which is described in detail.

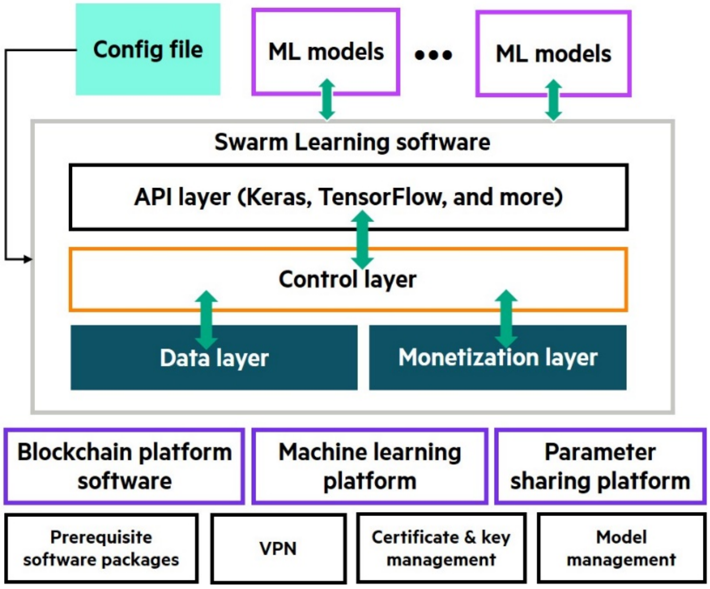
Config file
ML models
•••
ML models
Swarm Learning software
API layer (Keras, TensorFlow, and more)
Control layer
Data layer
Monetization layer
Blockchain platform software
Machine learning platform
Parameter sharing platform
Prerequisite software packages
VPN
Certificate & key management
Model management

FIGURE 2. Framework architecture

API layer

Swarm Learning is implemented as an API library available for multiple popular frameworks such as TensorFlow, Keras, and such. These APIs provide an interface that is similar to the training APIs in the native frameworks familiar to the data scientists. Calling these APIs automatically inserts the required hooks for Swarm Learning so that nodes seamlessly exchange parameters at the end of each model training epoch, and subsequently continue the training after resetting the local models to the globally merged parameters. With a few simple code changes, the entire network learns as one cohort, with all the complexities of control and data flow taking place under the hood.

Control layer

The responsibility for keeping the decentralized Swarm network in a globally consistent state lies with the control layer, and it is implemented using blockchain technology. The layer ensures that all operations and the corresponding state transitions are performed in an atomic manner. Both state and supported operations of the system are encapsulated in a blockchain smart contract.

The state comprises information such as the current epoch, the current members of the Swarm with their IP addresses and ports, and URIs for parameter files. The set of operations includes the logic to elect the leader of the Swarm toward the end of each epoch, fault-tolerance, and self-healing mechanisms, along with signaling among nodes for commencement and completion of various phases.

Data layer

The data layer takes charge of reliable and secure sharing of model parameters across the Swarm network. Like the control layer, this layer is also pluggable, supporting different file-sharing mechanisms such as HTTPS/TLS, IPFS, and such. The layer is controlled through the operations invoked by the control layer where the information about this layer is also maintained.

Monetization layer

This layer meters data usage and contribution during the model training process to calculate the monetary rewards for each of the participants, which are dispensed at the end of the training. It relies on blockchain's tamperproof and self-verifying smart contract to keep track of the contributions, and the built-in cryptocurrency framework to transfer rewards in a fully automated fashion.

HOW WILL SWARM LEARNING BENEFIT YOUR BUSINESS?

Swarm Learning started with the purpose of solving the dilemma of the explosion of data and the technical, social, and economic challenges of extracting values from data. Enterprises embracing Swarm Learning will benefit from the following competitive advantages:

Efficiency

Insights embedded in data do not come free of charge. In particular, the centralized ML approach, when aggregating data to a central location for processing, faces the costs of data transfer and investment in the storage and computing capability for that consolidated data. Swarm Learning eliminates the costs of raw data transfer by performing learning at or near the data sources. Our experiments show that the model parameters transferred can be more than 1000X smaller than the raw data. This tremendously lowers the data transfer costs and delays.

Swarm Learning can further reduce operation costs by taking advantage of existing storage and computing capabilities at or near the data sources, thus eliminating the investment in a central facility, either on-premises or in a cloud, for centralized storage and processing of the aggregated data, which can grow quickly in a centralized ML approach.

Privacy and security compliance

The enactment and enforcement of GDPR mark a critical milestone in legislative efforts on data privacy protection. Across the globe, businesses are under stricter scrutiny in transparency and accountability of their customer data-handling practice. The consequence of not having regulation compliance can hurt brand image and incur hefty fines. A large social media company, for example, recently reached an unprecedented $5 billion settlement with the U.S. government over its user privacy violation.[14]

Swarm Learning helps businesses comply with privacy and security regulations by giving data owners greater control over access to and usage of their data through the smart contract of blockchain, as well as eliminating the need for raw data transfer. Strong compliance with security and privacy regulations boosts customer confidence, and in turn, brings a business more revenue.

Fault-Tolerance

Compared with the centralized learning approach, Swarm Learning decentralizes both data storage and learning, thus effectively avoiding a single point of failure, which threatens business continuity. The Swarm Learning algorithm is effective in handling biased and unbalanced data at the various sources, and the smart contract is robust in handling exceptions such as the lost connection of a data source to its Swarm Learning peers.

Timely insights

Swarm Learning brings with it the powerful benefit of reducing the latency between the creation of data and the availability of actionable insight derived from that data. With Swarm Learning, model retraining can be initiated as soon as new data becomes available at any data source. The learning captured can be shared immediately with all the Swarm Learning peers, without waiting for the data to be transferred, consolidated, and then mined. A shorter path between data and insights means faster and more accurate responses to the ever-changing market, an enviable competitive advantage.

[14] Facebook Agrees to Pay $5 Billion and Implement Robust New Protections of User Information in Settlement of Data-Privacy Claims, Department of Justice, July 2019

New collaboration and monetization models

By drawing a clear line between raw data and the insights embedded in that data, Swarm Learning decouples data access and data ownership. This decoupling, along with the shifting of computing to data, provides a holistic perspective on the value of data.

Imagine a world where personal or sensitive information never leaves a user or an organization's direct control. Any service relying on these data is containerized, authorized by the data owner, and brought to the data, with a contract clearly stating data exposure, usage patterns, contract duration, and expected rewards for the data owner. This shifts control over personal or sensitive information to the data owners and opens a whole new model for data collaboration and monetization. Without raw data ever leaving any personal devices or organization, ingeniously designed incentives will spark business models to facilitate data collaboration for both consumer and enterprise applications.

WHERE CAN WE APPLY SWARM LEARNING?

Swarm Learning is more than a remedy for the challenges facing the centralized ML approach; it is also a powerful tool to unlock the undervalued opportunities residing with distributed data and the computing power at or near them. Swarm Learning is designed to apply to the widest industries imaginable and we will demonstrate its merits through a few selected use cases as follow:

Healthcare

Imagine we have three breast cancer research institutes in the U.S., Europe, and Asia, each of which has its own limited and proprietary breast cancer data set. These research institutes share the same objective to improve breast cancer diagnosis by developing and training an ML model. Since each institute's data is limited in size and may have distribution bias toward particular demographics, they would like to share their data in a manageable way, ideally limited to breast cancer-related insight only and nothing else.

Such goodwill to cooperate and advance the wellbeing of entire human race, however, faces significant regulation compliance risks in the centralized ML approach. With existing regulations on medical records in their respective countries, simply obtaining the approval to share the raw data and to transfer it to a central location, potentially out of the country, could be an insurmountable roadblock.

With Swarm Learning, however, the regulation compliance risks are minimized by eliminating raw data transfer. Computation is brought to the data instead. A smart contract between the research institutes details and enforces how the insights from their respective data set will be extracted, shared, and rewarded. The collaboration between them is clearly defined, strictly enforced, and reliably tracked. The institutes now can focus more on the fundamental research collaboration instead of the operational overhead.

Urban mobility

Urbanization is a global megatrend. Coming with the increasingly densely populated urban areas are the growing pains of deteriorating traffic. Ride-hailing, autonomous driving, connected cars, and smart cities are all efforts that help address the mobility challenges from disparate angles. Solving the urban traffic burden requires a comprehensive approach incorporating a wide range of information including people's commuting routes, daily routine, road and weather information, and public and private events. An intelligent model encompassing all this information could lead to better traffic pattern prediction and potentially optimized transportation resource planning. How can all this information be consolidated for the centralized ML in a timely and cost-effective manner? The chances are slim, if possible at all.

Take one step back, even if the information is readily available for the centralized ML approach, do we capture the problem well enough for an optimized and efficient solution? We believe something fundamental is missing here to paint the full picture—people's flexibility in choice-making and a reward mechanism that we can design to influence their choices and to improve the overall system performance.

For example, when the highway becomes congested in the evening rush hours, would you be willing to leave your office ten minutes later than usual to lessen the traffic? Probably not, you might think. Now, what if that choice comes with free use of the tollway for the next morning, which can save you 20 minutes in commute? If the reward could sway you into accepting the offer, imagine how the traffic will be shaped if choices like this can be customized to millions of urban residents in a real-time and decentralized manner. This is exactly what Swarm Learning can enable.

Swarm Learning, with its decentralized architecture and reward system, is exactly what we need to solve complex system problems with human or organizational interactions. Not only does it address privacy and security concerns in sharing data, but it can also bring in and shape the individual or organizational behavior. Compared with the current intelligent system still focusing on modeling and prediction, Swarm Learning can close the loop and bring intelligent systems to a whole new level to learn, act, and evolve.

Collaboration in deep space

Swarm Learning has unparalleled advantages for applications where moving and consolidating a large amount of data for centralized learning is prohibitive in terms of latency and costs. Nothing can demonstrate this merit of Swarm Learning better than outer space collaboration, where data sources for ML are widely distributed and far from effective central coordination.

As we venture deeper into the universe, large amounts of data will follow in our footsteps. The challenges of learning from data sources separated at this distance scale go beyond the reliability, availability, bandwidth, and costs of the communication channels. Latency is a fundamental barrier that we cannot cross. As an example, the return latency between Mars and Earth can be as high as 40 minutes. Such a huge latency could easily make the difference between a successful and failed space mission. During a space fleet's journey to Mars, it is much more practical to use Swarm Learning between the spacecraft, which will be relatively close, rather than streaming the data back to earth with tremendous latency.

CONCLUSION

In today's digital economy, the ability to quickly and accurately act upon data is a critical advantage. Swarm Learning, by combining decentralized machine learning with blockchain technology, empowers enterprises to shorten the data-to-action delay cost-effectively and robustly. The adoption of Swarm Learning will spawn new collaboration and monetization opportunities and we, at Hewlett Packard Enterprise, have committed ourselves to explore this new frontier with our partners and customers.

Authors biographies

- **Rongliang (Leon) Zhou**, Strategy and Business Development Manager. Leon drives company-wide strategic initiatives and turns cutting-edge technologies into business opportunities.
- **Vishesh Garg**, Research Engineer. Vishesh researches, validates, incubates, and productizes emerging technologies.

LEARN MORE AT

hpe.com/us/en/insights/articles/swarm-learning-and-the-artificially-intelligent-edge-1908.html

a50000344ENW, February 2020, Rev. 1