Integrating AI and Blockchain: Developing AI Standards for Cardano

Consensus Mechanism

UnboundedMarket Team

February 17, 2025

Contents

I. Motivation

Integrating artificial intelligence (AI) into blockchain ecosystems can greatly improve decentralized platforms. In the Cardano ecosystem, AI is increasingly used for data analytics, domain-specific chatbots, and automated smart contract vulnerability detection. However, widespread adoption is hindered by the lack of standardized frameworks for AI integration. Without clear guidelines, developers struggle to create secure, efficient, and interoperable AI solutions that align with blockchain's decentralized principles. To address this, we develop an AI-blockchain integration framework built on four pillars: an API, certification, benchmarking, and consensus mechanisms.

Building on our previous work on API, model certification, and benchmarking procedures, this paper outlines our approach to developing an AI consensus mechanism. In our framework, AI consensus is a foundational pillar for decentralized AI systems, as it enables multiple models to collaboratively determine a final output through transparent, robust, and verifiable aggregation. In blockchain ecosystems—where decisions can have significant real-world impact—achieving consensus among AI agents is critical for ensuring reliability, fault tolerance, and resistance to adversarial influences.

Our proposed consensus mechanism leverages a modality-specific aggregation strategy to combine the collective expertise of various AI models. This is necessary to ensure a consensus mechanism that is applicable to AI models across different output modalities. For example, while discrete outputs may simply be aggregated and combined via weighted majority voting, textual outputs from multiple models can be refined via a distillation-based reranking process that synthesizes all candidate responses. We thus aim to deliver a general consensus mechanism that enhances trust, accountability, and performance in AI-driven blockchain applications.

We start by outlining related work in Section II followed by the technical background in Section III. In Section IV, we present our consensus mechanism, which we further discuss in Section V.

II. Related Work

Traditionally, consensus mechanisms play a fundamental role in blockchains by ensuring that distributed participants agree on a unified and tamper-resistant version of the ledger. The first consensus algorithms, such as Proof of Work (PoW) [1, 2] and Proof of Stake (PoS) [3], have been widely used to maintain the integrity of decentralized systems. These mechanisms provide security, prevent double-spending, and enable trustless coordination across a network of nodes.

Beyond their application in cryptocurrency networks, consensus mechanisms play a crucial role in decentralized artificial intelligence (AI) systems, where multiple models or agents collaborate to solve complex tasks. In particular, blockchain technology offers a promising approach to ensuring consensus among AI models and their outputs. Recent research has explored ensemble-based strategies for improving the performance of large language models (LLMs) [4, 5]. These strategies combine multiple models to leverage their collective strengths, enhancing accuracy, robustness, and generalization across various tasks.

The use of blockchain technology can enhance ensemble-based AI systems by providing a decentralized, immutable ledger to record model predictions and decisions. This ensures transparency, auditability, and traceability throughout the decision-making process [6, 7]. By integrating blockchain with AI model ensembles, it becomes possible to establish verifiable accountability for each model's contribution, preventing tampering and increasing trust in AI-driven outcomes.

Moreover, blockchain can be instrumental in AI model certification and benchmarking, as discussed in our previous research. Certification mechanisms enable models to be evaluated based on standardized criteria, providing a trust layer that ensures only reliable models contribute to ensemble decisions. Benchmarking can further refine these evaluations by continuously assessing model performance on diverse datasets and real-world tasks. By leveraging certified performance metrics, blockchain-based AI frameworks can assign weight to each model's contributions proportionally to its demonstrated reliability, thereby optimizing ensemble accuracy while mitigating the impact of low-quality or adversarial models.

A critical challenge in decentralized AI systems is managing Byzantine faults—situations where models fail unpredictably or behave maliciously. Blockchain-based consensus mechanisms can enhance Byzantine fault tolerance by ensuring that unreliable or adversarial models do not unduly influence decision-making. Strobel et al. [8, 9] highlight the use of blockchain technology to improve resilience against adversarial behaviors, reinforcing the necessity of robust consensus mechanisms that can sustain system integrity even in the presence of compromised agents.

Looking forward, integrating blockchain into decentralized AI ecosystems could pave the way for more secure, autonomous, and scalable AI collaboration. Future advancements in consensus algorithms, such as Proof of Contribution (PoC) or Federated Byzantine Agreement (FBA), may further refine AI model selection, reward mechanisms, and fault tolerance in distributed learning environments. These innovations will be key to unlocking the full potential of decentralized AI while maintaining trust, security, and efficiency.

III. Technical Background

The integration of blockchain and artificial intelligence (AI) requires a thorough understanding of the fundamental principles underlying both technologies. Here, we describe the technical background, including blockchain architecture, consensus mechanisms, ensemble learning, and Byzantine fault tolerance.

A. Blockchain Fundamentals

Blockchain is a decentralized, append-only ledger in which transactions are recorded in a sequence of blocks, each cryptographically linked to its predecessor [2]. Formally, let B_t represent the block at index t in the blockchain. Each block consists of:

- A set of transactions $T_t = \{T_{t,1}, T_{t,2}, ..., T_{t,n}\}$, where $T_{t,i}$ represents the i-th transaction in block t.
- A cryptographic hash $H(B_{t-1})$ of the previous block, ensuring immutability.
- A nonce N_t used for consensus validation (in PoW-based systems).
- A timestamp τ_t to order transactions chronologically.

Mathematically, the hash of a block can be represented as:

$$H(B_t) = H(H(B_{t-1})||T_t||N_t||\tau_t) \quad (1)$$

where $H(\cdot)$ is a cryptographic hash function, and $||$ denotes concatenation.

The blockchain network maintains an ordered ledger $\mathcal{B} = \{B_0, B_1, ..., B_t\}$, where each new block satisfies the validity conditions imposed by the consensus mechanism.

B. Consensus Mechanisms

Consensus mechanisms ensure agreement among distributed participants on the state of the blockchain [10]. These protocols prevent double-spending and maintain the security of decentralized systems. Let N be the number of nodes in the network, and let V be the set of validators participating in consensus ($V \subseteq N$).

a) Proof of Work (PoW): In PoW, miners solve a computational puzzle to append a new block. This is achieved by finding a nonce N_t such that the hash satisfies a difficulty constraint:

$$H(B_t) < D \quad (2)$$

where D is the current difficulty level. The expected number of trials for a successful solution is $O(2^d)$, where d is the number of leading zero bits required in the hash output [1].

b) Proof of Stake (PoS): In PoS, the probability $P(i)$ of a node i being selected to propose the next block is proportional to its stake s_i [3]:

$$P(i) = \frac{s_i}{\sum_{j \in V} s_j} \tag{3}$$

where s_i represents the number of tokens held by node i.

c) Federated Byzantine Agreement (FBA): FBA achieves consensus through quorum slices, where each node selects a subset of trusted participants [11]. A transaction is finalized when it receives signatures from a quorum Q satisfying:

$$Q_i \cap Q_j \neq \emptyset, \quad \forall i, j \in N \tag{4}$$

ensuring that consensus groups overlap sufficiently to prevent forks.

C. Ensemble Learning in AI

Ensemble learning improves performance by aggregating predictions from multiple models. Given M models $f_1, f_2, ..., f_M$, the ensemble prediction $F(x)$ is typically computed as a weighted combination:

$$F(x) = \sum_{i=1}^{M} w_i f_i(x) \tag{5}$$

where w_i represents the weight assigned to model f_i, subject to $\sum_{i=1}^{M} w_i = 1$.

a) Bagging: Instead of aggregating the outputs of multiple models, bagging constructs multiple instances of the same model on bootstrapped subsets of the training data [12].

b) Boosting: Boosting improves weak models iteratively by assigning higher weights to misclassified samples [13]. Given an initial uniform distribution $p_1(i) = \frac{1}{n}$, the weight update rule in AdaBoost is:

$$p_{t+1}(i) = p_t(i) \frac{e^{-\alpha_t y_i f_t(x_i)}}{Z_t} \tag{6}$$

where Z_t is a normalization factor, and α_t is the model's confidence.

c) Stacking: Stacking uses a meta-model $g(\cdot)$ to learn an optimal combination of base model predictions [14]:

$$F(x) = g(f_1(x), f_2(x), ..., f_M(x)) \tag{7}$$

D. Byzantine Fault Tolerance in Distributed AI

In decentralized AI systems, adversarial models can manipulate results [8]. Byzantine fault tolerance ensures resilience under such conditions.

Let $f : X \to Y$ be an AI model and $\mathcal{F} = \{f_1, f_2, ..., f_M\}$ be an ensemble of M models, where up to f_b models are Byzantine (malicious). The system achieves Byzantine fault tolerance if:

$$\sum_{i \in \text{honest}} w_i f_i(x) \gg \sum_{j \in \text{Byzantine}} w_j f_j(x) \tag{8}$$

for all $x \in X$, ensuring that honest models dominate decision-making.

IV. AI Consensus Mechanism

Building on our previous research, we start by outlining the requirements for AI consensus mechanisms.

A. Requirements

As outlined in the research proposal, consensus in decentralized AI systems requires three critical components: transparency and traceability in both models and their outputs, resilience to adversarial or faulty participants, and the integration of verified performance metrics to optimize decision-making. Transparency and traceability ensure that model outputs can be audited, leading to trust and accountability in the system. Resilience mechanisms mitigate the influence of unreliable or malicious actors, preserving the integrity of the consensus process. Furthermore, incorporating validated performance data allows the system to dynamically weight individual models based on their demonstrated reliability and predictive accuracy, improving the robustness and efficacy of ensemble-based decision strategies.

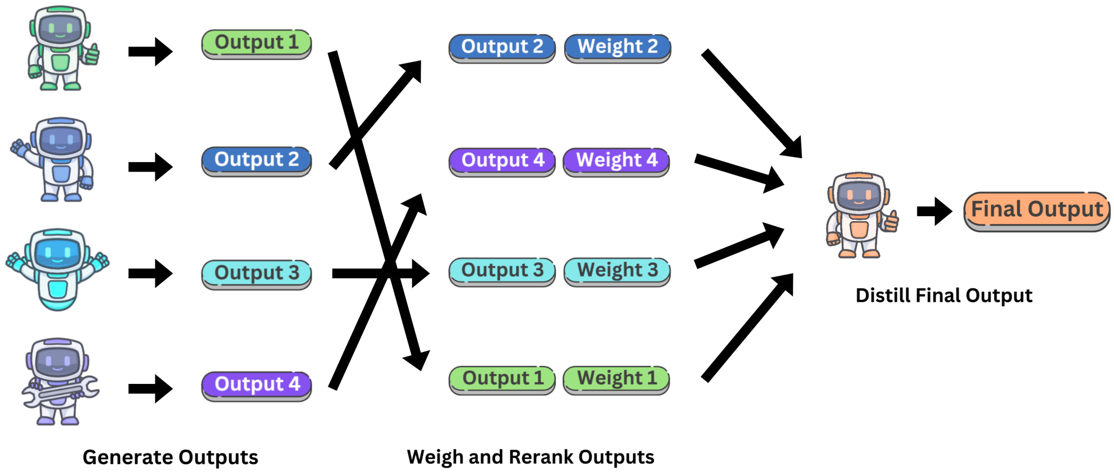

Fig. 1. Overview of the AI consensus mechanism: individual models first generate candidate outputs, which are then assigned weights based on certification criteria and benchmarking metrics. These weighted outputs are subsequently reranked, and an additional distillation model integrates the candidates and their weights to produce a refined final output.

From a formal perspective, let $\mathcal{M} = f_1, f_2, ..., f_M$ represent the set of AI models participating in consensus. Each model f_i is characterized by a set of parameters θ_i, training data D_i, and performance history P_i. The consensus function $C(\mathcal{M})$ must satisfy the following properties:

- **Verifiability:** All models must have transparent and immutable records stored on-chain.
- **Fault tolerance:** The consensus mechanism must be robust to Byzantine failures, ensuring that malicious models do not significantly impact the final decision.
- **Adaptability:** Weights assigned to models must be dynamically adjusted based on their historical accuracy and reliability.
- **Scalability:** The framework must be efficient even as the number of participating models grows.

These requirements ensure that AI-driven decision-making remains trustworthy and computationally feasible in a decentralized system.

B. API, Certification and Benchmarking

Our approach to AI consensus builds on our foundational research in API frameworks, certification, and benchmarking.

1) NFT-Based Model Cards: A fundamental component of our framework is the introduction of NFT-based model cards, which encode important metadata for each model. Our model cards include details such as the architecture and domain of the respective model, parameter counts, or training data sources.

Each model f_i is associated with a model card MC_i, which is hashed and stored on-chain:

$$H(MC_i) = H(M_1||M_2||\cdots||M_n), \qquad (9)$$

where H is a cryptographic hash function, $||$ denotes concatenation, and $M_1, M_2, \ldots, M_n$ represents the individual components of the model card, such as architecture, domain, parameter counts and training data. Our model cards standardize model documentation and provide an immutable artifact that can be used to audit the model.

2) Certification and Benchmarking: Before models are admitted into the consensus process, they undergo a rigorous certification process. To do so, standardized benchmarking tests produce a performance score for each model:

$$B(f_i) = \sum_{j=1}^{n} w_j, \delta(f_i(x_j), y_j), \qquad (10)$$

where:

- $\delta(a, b)$ is the Kronecker delta function,
- w_j is the weight associated with test case j, and
- (x_j, y_j) are the test inputs and corresponding ground truth labels.

Together with certification status $C(f_i)$ (with $C(f_i) = 1$ if certified, 0 otherwise), these benchmarking results are permanently recorded on-chain.

C. Categorization of AI Models by Output Modality

To effectively apply a consensus mechanisms, we first need to categorize AI models based on their output modality. This classification allows for tailored aggregation strategies that respect the distinct nature of each model type:

- **Discrete**: Models that perform classification or structured decision-making, such as Text Classification, Sentiment Analysis, Anomaly Detection, Spam Detection, Toxicity Detection, Image Classification, and Time-Series Forecasting.
- **Text**: Models that generate textual outputs, including Conversational AI, Visual Question Answering (VQA), and Image Captioning.
- **Vision**: Models that generate visual outputs, including Image Generation and Video Generation.
- **Audio**: Models that generate audio outputs, such as Text-to-Speech (TTS) and Music Generation.

Each class requires different techniques to ensure meaningful aggregation and consensus.

D. Weighted Model Consensus by Output Modality

For models in the eligible subset S, the influence on the final prediction is determined through a weighted consensus mechanism. The weight assigned to an eligible model f_i is given by:

$$W(f_i) = \frac{B(f_i) \cdot C(f_i)}{\sum_{j \in S} B(f_j) \cdot C(f_j)}, \tag{11}$$

where $B(f_i)$ represents the model's performance and $C(f_i)$ accounts for the certification status. The consensus process is tailored to the aforementioned different output modalities. For a simplified overview, see Figure 1.

1) Discrete: Weighted Majority Voting: In the case of discrete outputs, when can exactly apply consensus as weighted majority voting, as given in Equation (5). Given outputs that belong to a finite set Y, we employ the following weighted majority voting:

$$\hat{y} = \sum_{i \in S} W(f_i) \cdot \delta(f_i(x), y), \tag{12}$$

where $\delta(f_i(x), y)$ is an indicator function evaluating whether model f_i predicts y. This ensures robustness by prioritizing high-confidence models that adhere to predefined structural and domain-specific constraints.

2) Text: Aggregation via Reranking.: Since textual outputs can be highly variable, we cannot simply combine the outputs of multiple models into a coherent output. Thus, simple majority voting is not applicable. Recent research has investigated several methods for combining the outputs of several LLMs into a final output [15, 16, 17, 18, 19]. However, in contrast to most approaches that merely select the highest-ranked candidate, we propose a distillation-based method to synthesize a final textual output, which most closely resembles the one from Jiang et al. [20]. We first generate outputs y_i for each model f_i, which are then paired with their weight $W(f_i)$. This results in a ranking of the available model responses. We then use a separate model, which takes in the pairs $(y_i, W(f_i))_{i=1}^{M}$ and produces a refined output.

Specifically, let $\{y_i\}_{i \in S}$ denote the candidate responses produced by models in the eligible set S, each associated with weight $W(f_i)$. Instead of choosing $\hat{y}$ as the single candidate with the highest rank, we define a distillation process:

$$D : \{(y_i, W(f_i))\}_{i \in S} \to \mathcal{Y},$$

which generates the final consensus output

$$\hat{y} = D\big(\{(y_i, W(f_i))\}_{i \in S}\big) \tag{13}$$

Here D is the aggregator model, which distills the final output $\hat{y}$. Although the aggregator model itself may not be as powerful as the individual candidate generators, its role is primarily to distill the shared knowledge among the candidates, thereby mitigating individual biases and errors.

3) Vision: Distillation via Latent Feature Aggregation: For visual outputs—including both static images and dynamic sequences—we propose a general distillation mechanism [21, 22]. Each model f_i provides a visual output, denoted by V_i, which may represent a single image or a sequence of frames. To capture the salient features, we first extract a latent representation using an encoder Φ:

$$\Phi(V_i) \in \mathcal{Z},$$

where $\mathcal{Z}$ is the latent feature space. Rather than directly reconstructing the final output via weighted fusion, we employ a distillation function

$$D_V : \{(\Phi(V_i), W(f_i))\}_{i \in S} \to \mathcal{F},$$

which aggregates the latent features from the candidate outputs into a unified latent representation $\mathcal{F}$. The final consensus visual output $\hat{V}$ is then reconstructed by a decoder Ψ:

$$\hat{V} = \Psi\Big(D_V\big(\{(\Phi(V_i), W(f_i))\}_{i \in S}\big)\Big). \tag{14}$$

While the precise architectures of Φ, D_V, and Ψ may vary based on the application (e.g., image synthesis or video generation), the overarching goal remains to distill the most salient and robust features from the candidate outputs. This approach leverages the complementary strengths of each individual model, producing a consensus output that is more coherent and representative of the collective expertise of the ensemble.

4) Audio: Distillation via Spectro-Temporal Feature Fusion: Ensemble learning for audio models is a particularly difficult discipline since the model outputs are highly subjective. We, therefore, divert towards rigorously determining the best-fitting model using standardized metrics. These metrics vary depending on the intended use and output a simple score between 0 and 1, where 0 is the lowest and 1 is the highest. For example, for the task of voice synthetization from text, clarity of speech and adherence to style guidelines are of key importance. For assessing clarity, common choices are the Word Error Rate (WER), Speech intelligibility Index (SII) [23], or Short-Time Objective Intelligibility (STOI) [24, 25]. The style of the speech can be assessed using psycho-acoustic metrics such as the Mel Cepstral Distortion (MCD) [26] or the Perceptual Objective Listening Quality Analysis (POLQA) [27]. Based on these objective quality indicators I, a ranking for each output would be created.

We then transform the audio signals of each model f_i into the frequency domain, yielding an audio spectrogram $A_i \in \mathbb{R}^{F \times T}$. This representation captures the temporal evolution of the frequency characteristics of the model output. Rather than directly averaging these spectrograms, we define a conditional distillation function:

$$D_A : \{(A_i, W(f_i)) | I_i\}_{i \in S} \to \mathcal{A},$$

which fuses the spectro-temporal features into an aggregated spectrogram representation $\hat{A} \in \mathcal{A}$ proportionally to the quality indicator I_i. The final audio waveform is then obtained via an inverse (short-time) Fourier transform:

$$\hat{a} = \mathrm{ISTFT}(\hat{A}). \tag{15}$$

Overall, the proposed distillation-based aggregation methods for text, video, and audio outputs share a common principle: rather than simply selecting or averaging candidate outputs; they synthesize a consensus output that contains the collective expertise of the individual models. Even if the final aggregator model possesses less capacity than the candidate generators, its ability to distill and integrate shared information results in outputs that are more consistent and reliable [20, 19].

V. Discussion

This section reflects on the requirements and our consensus approach, summarizing key insights and addressing challenges.

A. Summary

Our work presents a framework for AI consensus as a foundational pillar for integrating AI into the Cardano ecosystem, along with accessibility via an API, certification, and benchmarking. At the heart of our approach is a consensus mechanism that leverages a dynamic, weighted ensemble in which verifiable, on-chain metrics derived from its certification status and benchmarking results determine each model's influence. The consensus mechanism builds on our previous work on NFT-based model cards that guarantee transparency and traceability by immutably recording critical model metadata and supporting a rigorous certification process that reinforces accountability. Moreover, the framework incorporates an adaptive penalization strategy to mitigate the impact of adversarial or faulty models, thereby enhancing the robustness and reliability of the overall consensus. Additionally, the evaluation metrics within our framework are easily extensible, allowing continuous refinement over time. This ensures that our AI Consensus mechanism remains adaptable and responsive to evolving standards and data sources. Our AI Consensus mechanism thus provides a scalable and secure means of aggregating diverse model outputs, designed to ensure high-quality decision-making in a decentralized setting.

B. Challenges

Implementing our AI consensus mechanism presents several interrelated challenges. First, the heterogeneity of modalities—ranging from textual and visual outputs to audio signals and discrete decisions—necessitates the development of specialized aggregation strategies that can reconcile differing data structures and error profiles into a coherent final output. While we have addressed the most common output modalities, we cannot deal with all possible modalities.

Moreover, the dynamic weighting of models based on real-time performance and certification status demands sophisticated benchmarking methods that are resilient to fluctuations in model behavior and potential adversarial manipulation.

Scalability further complicates the design of these systems; as the number of participating models increases, so too does the computational overhead required for aggregating outputs and maintaining on-chain verifiability. In parallel, ensuring fault tolerance and security remains an important aspect, particularly in the face of Byzantine failures and malicious actors whose actions may compromise the integrity of the consensus process.

C. Future Work: Implementations

Our future research will focus on the practical implementation of the AI Consensus framework within the Cardano ecosystem. Addressing scalability will require the development of efficient data architectures that combine off-chain storage with on-chain data, thus maintaining transparency while mitigating storage overhead. In parallel, we plan to undertake several simulations to empirically validate the performance, robustness, and scalability of the consensus mechanism. These studies will allow us to fine-tune the dynamic weighting schemes and adaptive penalization parameters, ensuring that the system remains resilient against adversarial attacks under real-world conditions.

VI. CONCLUSION

In this paper, we have presented a detailed framework for achieving AI consensus within decentralized blockchain environments. Building on our prior work in API development, model certification, and benchmarking, we introduced innovative consensus mechanisms that integrate modality-specific aggregation strategies. Our approach is based on a weighted majority voting for discrete outputs, distillation-based reranking for textual outputs, and analogous latent feature and spectro-temporal distillation methods for visual and audio outputs, respectively. Our approach is designed to ensure that the final aggregated outputs are not only robust and fault-tolerant but also reflective of the collective expertise of the participating AI models. By embedding transparency, verifiability, and adaptability into the consensus process, our framework addresses key challenges inherent in AI-driven blockchain applications and paves the way for enhanced trust and reliability.

REFERENCES

[1] A. Back, "Hashcash - a denial of service counter-measure," 2002. [Online]. Available: http://www.hashcash.org/papers/hashcash.pdf

[2] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Cryptography Mailing list at https://metzdowd.com*, 03 2009. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[3] S. King and S. Nadal, "PPCoin: Peer-to-peer crypto-currency with proof-of-stake," 2012. [Online]. Available: https://decred.org/research/king2012.pdf

[4] D. Jiang, X. Ren, and B. Y. Lin, "LLM-blender: Ensembling large language models with pairwise ranking and generative fusion," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 14 165–14 178. [Online]. Available: https://aclanthology.org/2023.acl-long.792/

[5] F. Xue, Z. Zheng, Y. Fu, J. Ni, Z. Zheng, W. Zhou, and Y. You, "Openmoe: An early effort on open mixture-of-experts language models," 2024. [Online]. Available: https://arxiv.org/abs/2402.01739

[6] K. Salah, M. H. U. Rehman, N. Nizamuddin, and A. Al-Fuqaha, "Blockchain for ai: Review and open research challenges," *IEEE Access*, vol. 7, pp. 10 127–10 149, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8598784

[7] M. Shafay, R. W. Ahmad, K. Salah, I. Yaqoob, R. Jayaraman, and M. A. Omar, "Blockchain for deep learning: review and open challenges," *Cluster Computing*, vol. 26, pp. 197 – 221, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:239545552

[8] V. Strobel, E. Castelló Ferrer, and M. Dorigo, "Managing byzantine robots via blockchain technology in a swarm robotics collective decision making scenario," in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, ser. AAMAS '18. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2018, p. 541–549. [Online]. Available: https://ifmas.csc.liv.ac.uk/Proceedings/aamas2018/pdfs/p541.pdf

[9] V. Strobel, E. Castelló Ferrer, and M. Dorigo, "Blockchain technology secures robot swarms: A comparison of consensus protocols and their resilience to byzantine robots," *Frontiers in Robotics and AI*, vol. 7, 2020. [Online]. Available: https://www.frontiersin.org/journals/robotics-and-ai/articles/10.3389/frobt.2020.00054

[10] S. Bano, A. Sonnino, M. Al-Bassam, S. Azouvi, P. McCorry, S. Meiklejohn, and G. Danezis, "Sok: Consensus in the age of blockchains," in *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, ser. AFT '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 183–198. [Online]. Available: https://doi.org/10.1145/3318041.3355458

[11] D. Mazières, "The stellar consensus protocol: A federated model for internet-level consensus," 2015. [Online]. Available: https://johnpconley.com/wp-content/uploads/2021/01/stellar-consensus-protocol.pdf

[12] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, pp. 123–140, 1996. [Online]. Available: https://link.springer.com/article/10.1007/BF00058655

[13] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S002200009791504X

[14] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608005800231

[15] W. Fedus, B. Zoph, and N. Shazeer, "Switch transformers: scaling to trillion parameter models with simple and efficient sparsity," *J. Mach. Learn. Res.*, vol. 23, no. 1, Jan. 2022.

[16] W. Li, Y. Peng, M. Zhang, L. Ding,

H. Hu, and L. Shen, "Deep model fusion: A survey," 2023. [Online]. Available: https://arxiv.org/abs/2309.15698

[17] H. Hoang, H. Khayrallah, and M. Junczys-Dowmunt, "On-the-fly fusion of large language models and machine translation," in *Findings of the Association for Computational Linguistics: NAACL 2024*, K. Duh, H. Gomez, and S. Bethard, Eds. Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 520–532. [Online]. Available: https://aclanthology.org/2024.findings-naacl.35/

[18] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. de las Casas, E. B. Hanna, F. Bressand, G. Lengyel, G. Bour, G. Lample, L. R. Lavaud, L. Saulnier, M.-A. Lachaux, P. Stock, S. Subramanian, S. Yang, S. Antoniak, T. L. Scao, T. Gervet, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed, "Mixtral of experts," 2024. [Online]. Available: https://arxiv.org/abs/2401.04088

[19] J. Lu, Z. Pang, M. Xiao, Y. Zhu, R. Xia, and J. Zhang, "Merge, ensemble, and cooperate! a survey on collaborative strategies in the era of large language models," 2024. [Online]. Available: https://arxiv.org/abs/2407.06089

[20] D. Jiang, X. Ren, and B. Y. Lin, "LLM-blender: Ensembling large language models with pairwise ranking and generative fusion," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 14 165–14 178. [Online]. Available: https://aclanthology.org/2023.acl-long.792/

[21] Y. Duan, J. Zhang, and L. Zhang, "Dataset distillation in latent space," 2023. [Online]. Available: https://arxiv.org/abs/2311.15547

[22] B. B. Moser, F. Raue, S. Palacio, S. Frolov, and A. Dengel, "Latent dataset distillation with diffusion models," 2024. [Online]. Available: https://arxiv.org/abs/2403.03881

[23] A. S. of America. Standards Secretariat, A. N. S. Institute, A. S. of America, and B. Accredited Standards Committee S3, *American National Standard: Methods for Calculation of the Speech Intelligibility Index*. Acoustical Society of America, 1997. [Online]. Available: https://books.google.ch/books?id=LdO4PwAACAAJ

[24] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "A short-time objective intelligibility measure for time-frequency weighted noisy speech," in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2010, pp. 4214–4217. [Online]. Available: https://ieeexplore.ieee.org/document/5495701

[25] ——, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011. [Online]. Available: https://ceestaal.nl/Taal%282011%29.pdf

[26] X. Xu, C. Pautasso, L. Zhu, V. Gramoli, A. Ponomarev, A. B. Tran, and S. Chen, "The blockchain as a software connector," in *2016 13th Working IEEE/IFIP Conference on Software Architecture (WICSA)*, 2016, pp. 182–191. [Online]. Available: https://ieeexplore.ieee.org/document/7516828

[27] Y. Gaoxiong and Z. Wei, "The perceptual objective listening quality assessment algorithm in telecommunication: Introduction of ITU-T new metrics POLQA," in *2012 1st IEEE International Conference on Communications in China (ICCC)*, 2012, pp. 351–355. [Online]. Available: https://ieeexplore.ieee.org/document/6356906