Integrating AI and Blockchain: Developing AI Standards for Cardano

Standards for API, Certification and Benchmarking

UnboundedMarket Team

December 18, 2024

Contents

I. Motivation

Integrating artificial intelligence (AI) into blockchain ecosystems has the potential to significantly improve the capabilities of decentralized platforms. Within the Cardano ecosystem, an increasing number of projects have started using AI for various applications, including data analytics, domain-specific chatbots, and automated identification of vulnerabilities in smart contracts. These use cases underscore the significant value of AI for blockchain ecosystems.

Despite the promise of AI-driven blockchain solutions, a critical barrier to widespread adoption lies in the absence of standardized frameworks for AI integration. Without clear guidelines and protocols, developers face significant challenges in designing secure, efficient, and interoperable AI solutions that align with the decentralized principles of blockchain systems.

We address this gap by defining a cohesive framework for AI-blockchain integration, grounded in four foundational pillars: an API framework (Section II), a Certification Procedure (Section III), a Benchmarking Procedure (Section IV), and Consensus Mechanisms (discussed in Section V). Our proposed framework not only addresses technical challenges but also promotes trust, accountability, and comparability, key elements for driving innovation in the Cardano ecosystem.

Building on our previous research proposal, this paper outlines concrete plans for developing three of the four foundational pillars: the API framework, the certification procedure, and the benchmarking procedure. By introducing standardized protocols, robust certification mechanisms, and transparent benchmarking processes, we aim to accelerate the development and adoption of AI-powered solutions on Cardano, setting a strong precedent for the broader blockchain community.

II. API Framework for Integrating AI Models with Blockchain

Integrating AI models into blockchain ecosystems, such as Cardano, requires a robust API framework. APIs facilitate interaction between decentralized applications (dApps) and AI services by defining standardized protocols and communication rules. These standards ensure accessibility, scalability, and reliability of services, enabling seamless integration and reducing barriers to adoption.

In this section, we outline an API framework to address the unique requirements of blockchain-based AI integration. The framework incorporates insights from existing AI-Service APIs, identifies gaps in blockchain compatibility, and introduces innovative features to support decentralized systems.

A. Essential Features of the API Framework

We start by outlining the essential features of the API framework.

a) Standardized Communication Protocols: RESTful APIs are chosen as the foundational architecture due to their simplicity, scalability, and widespread adoption in web-based services [1, 2, 3]. Following best practices outlined in the OpenAPI Specification (OAS), the proposed API structure adheres to standardized principles, ensuring reliable communication between dApps and AI services.

b) Blockchain Integration: Existing AI APIs lack direct blockchain integration. To address this, the API introduces routes that interact with smart contracts, enabling on-chain recording of transactions, certifications, and benchmarking results. Smart contracts enforce usage terms, manage fees, and validate transactions, ensuring transparency and accountability.

c) Scalability and Efficiency: The framework supports asynchronous processing, load balancing, and batch inference capabilities to handle high transaction volumes. These features ensure minimal latency and efficient resource utilization, even under heavy network traffic.

d) Developer Support and Documentation: Extensive documentation, generated from the OpenAPI specifications, provides developers with clear guidance on implementation [4, 5]. SDKs for popular programming languages enable rapid adoption and simplify integration with blockchain ecosystems. Adherence to the OpenAPI standard enables developers to automatically validate the

API specification and derive server- and client-side code from it.

B. API Structure

The API is categorized into two main groups of routes: *Administration Routes* and *Inference Routes*.

1) Administration Routes: Administration routes enable the setup and management of smart contracts, and facilitate smooth and efficient execution of AI inferences:

- **registerInferenceService:**
  Purpose: Registers a new inference service.
  Arguments:
  - *endpoint_id:* Identifier for the AI model endpoint.
  - *num_queries:* Number of intended queries.
  - *userWalletAddress:* Blockchain wallet address of the user.
  - *is_revisable (optional):* Indicates if the contract can be terminated early.
  - *is_modifiable (optional):* Allows for changes in contract terms.

  Return:
  inferenceServiceId: Unique identifier for the registered service.
- **revokeInferenceService:**
  Purpose: Terminates an inference service prematurely, if permitted.
  Arguments:
  - *inferenceServiceId:* Identifier of the inference service.
  - *userWalletAddress:* Blockchain wallet address of the user.
- **updateInferenceService:**
  Purpose: Modifies an existing inference service (e.g., adding or withdrawing funds).
  Arguments:
  - *inferenceServiceId:* Identifier of the inference service.
  - *userWalletAddress:* Blockchain wallet address of the user.

2) Inference Routes: Inference routes interact directly with AI models and smart contracts, providing mechanisms for executing AI queries:

- **singleInference:**
  Purpose: Executes a single AI inference query.
  Arguments:
  - *endpoint:* URI or identifier of the AI model.
  - *query:* Input data for the AI model.
  - *inferenceServiceId:* Identifier for the associated smart contract.
  - *optionalParameters:* Additional parameters for model-specific customization.
- **batchInference:**
  Purpose: Executes multiple inference queries in a batch.
  Arguments:
  - *endpoint:* URI or identifier of the AI model.
  - *queryList:* List of input queries.
  - *inferenceServiceId:* Identifier for the associated smart contract.
  - *optionalParameters:* Additional parameters for model-specific customization.

C. Implementation Workflow

The implementation of the API follows a structured workflow:

1) **Defining API Specifications:** The API routes and properties are specified in OpenAPI format (YAML/JSON). The specifications include mandatory and optional parameters for each route.
2) **Validating the Specification:** Running the API specification through a validator ensures consistency and smooth deployment.
3) **Generating Server-Side Code:** Based on the specifications, server-side code is generated implementing the Rest-API server-side.
4) **Generating Client-Side Code (SDK):** Client libraries enable community developers to consume the API and stay up-to-date with any potential changes.

The proposed API framework establishes a standardized, scalable, and blockchain-compatible interface for AI model integration. Leveraging smart contracts ensures secure and transparent interactions between users, AI services, and the blockchain. This approach not only aligns with best practices in API design [6] but also addresses the unique challenges of decentralized systems, paving the way for widespread adoption of AI in blockchain ecosystems.

III. Certification

Certification is a critical component in establishing trust, accountability, and transparency in using AI models. By verifying compliance with predefined standards, certification ensures that AI systems meet functional, ethical, and operational criteria [7, 8]. This section outlines the theoretical foundation for a robust certification procedure tailored to AI models, emphasizing standardized documentation and blockchain integration for enhanced transparency and traceability.

A. Certification Parameters

The proposed certification process emphasizes creating standardized documentation for AI models, encapsulated in the form of model cards. Model cards serve as a documentation framework that provides detailed information about an AI model's capabilities, limitations, training data, and intended use cases [9]. Model cards are already widely used, for example, in IBM's Watson.ai, which offers model cards that include detailed model parameters and usage guidelines[1].

1) Fixed Parameters: To ensure consistency and comparability across models, the certification framework defines a set of mandatory parameters. These parameters include:

- **Usage Instructions:** A clear description of how to implement and interact with the model.
- **Associated Costs:** Information about financial or computational costs related to the deployment and use of the model.
- **Model Size:** The number of parameters, indicating the scale and complexity of the model.
- **Token/Usage Limits:** Restrictions or thresholds for model usage, if applicable.
- **Instruction tuning:** Details on fine-tuning with instruction-following data.
- **Model Architecture:** Details about the underlying architecture, such as Transformer-based models [10] or recurrent neural networks [11].
- **Licensing Terms:** Explicit licensing information that dictates how the model may be used, distributed, or modified.

2) Optional Parameters: Optional parameters provide additional context for specific models or use cases. These may include:

- **Training Data Sources:** A description of data sets used for model training that offers insights into the provenance of the data and potential biases.
- **Ethical Considerations:** Additional information on fairness, bias mitigation, and other ethical concerns related to the deployment of the model.
- **Intended Use Cases and Limitations:** A clear delineation of scenarios where the model is expected to perform effectively, as well as known limitations or risks in its application. This ensures users understand the appropriate scope of the model's functionality.

B. Blockchain-Integrated Certification

To enhance trust and transparency, the certification process can be integrated within a blockchain-based infrastructure. Using platforms like Cardano ensures immutability, decentralized verification, and improved traceability [12, 13, 14]. The inherent properties of the blockchain provide several advantages for the certification of the AI model:

- **Immutability and Version Tracking:** Certification records stored on the blockchain are immutable and can include real-time updates and version histories. This is essential for AI models that evolve rapidly, requiring frequent updates to their certifications.
- **Decentralized Verification:** Blockchain allows for distributed and consensus-based verification, reducing reliance on centralized authorities.

[1] https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/fm-models.html?context=wx

- **Traceability and Auditability:** The transparent and traceable nature of blockchain ensures that certifications are auditable, supporting continuous compliance with evolving standards and regulations.
- **NFT-Based Model Cards:** To further leverage blockchain technology, certification documentation, such as model cards, can be minted as Non-Fungible Tokens (NFTs). Minting model cards as NFTs ensures their uniqueness, immutability, and verifiability while enabling seamless sharing, distribution, and proof of authenticity across stakeholders.

This blockchain-based certification approach supports the continuous evolution of AI models by maintaining an auditable and verifiable history of their compliance. The addition of NFTs for model cards introduces a new level of accessibility and accountability, allowing certified AI models to be transparently showcased or exchanged. By doing so, this method strengthens comparability across models and provides a robust foundation for subsequent benchmarking efforts (see Section IV).

IV. Benchmarking

Benchmarking is an important aspect of AI model evaluation, providing systematic methods to assess model performance across various dimensions such as robustness, efficiency, and scalability. The benchmarking process enables comparative performance analysis and thereby offers valuable insights for selecting models tailored to specific applications. This section outlines the theoretical foundation for robust benchmarking procedures and proposes a concrete benchmarking process that integrates blockchain technology to enhance transparency, reliability, and accountability in AI evaluation.

A. Benchmarking for AI Models

AI benchmarking typically involves evaluating models on a standardized set of tasks, using predefined datasets and metrics. Benchmarks are designed to assess specific capabilities, such as visual recognition in ImageNet [15] or language understanding in GLUE (General Language Understanding Evaluation) [16]. These benchmarks have played an important role in advancing AI research by enabling reproducible comparisons and driving progress.

However, conventional benchmarking practices often exhibit limitations:

- **Static and Narrow Focus:** Traditional benchmarks rely on static datasets and narrowly defined tasks, which may not capture the complexity of real-world applications [17].
- **Limited Domain Coverage:** Benchmarks often focus on specific domains, neglecting generalizability across diverse use cases.
- **Transparency Challenges:** The growing trend of closed-source models (e.g., OpenAI's proprietary systems) hinders reproducibility and independent verification of benchmark results.

To address these challenges, recent efforts have introduced holistic evaluation suites like HELM (Holistic Evaluation of Language Models) [18] and the Language Model Evaluation Harness [19, 20], which combine diverse tasks and metrics. These approaches aim to provide a more comprehensive assessment of AI capabilities but still face limitations in transparency and reproducibility when applied to closed-source systems.

B. The Role of Blockchain in Benchmarking

Integrating blockchain technology into the benchmarking process introduces novel solutions to transparency and reliability challenges. Blockchain's inherent properties of immutability, decentralized verification, and traceability make it an ideal medium for recording and sharing benchmark results in a secure and tamper-proof manner. Specifically, blockchain-based benchmarking offers the following advantages:

- **Immutable Records of Performance:** Benchmark results stored on-chain ensure that evaluations are tamper-proof and auditable, maintaining the integrity of performance claims.
- **Enhanced Transparency:** By jointly storing model cards and benchmark results on the blockchain (see Section III), stakeholders can

access verifiable details about model architecture, training data, and evaluation protocols, even for closed-source models.
- **Reproducibility and Trust:** Blockchain facilitates decentralized verification of benchmark claims, enabling independent parties to reproduce evaluations or verify results without relying on the model provider's transparency.

C. Requirements for Benchmarking Procedures in Blockchain Environments

To establish a robust benchmarking framework integrated with blockchain, the following requirements must be addressed:

a) Diverse and Comprehensive Evaluation Tasks: Benchmarking must include tasks that assess a wide range of capabilities across different domains (e.g., vision, language, multi-modal tasks). This diversity ensures broad applicability of benchmarks and mitigates the risk of overfitting models to specific tasks.

b) On-Chain Storage of Results and Metadata: To enhance accountability, the benchmarking procedure should include storing both evaluation results and metadata (e.g., task descriptions, evaluation metrics, dataset details) on the blockchain. Joint storage with model cards ensures that stakeholders can trace performance back to specific model versions and evaluation protocols, promoting end-to-end transparency.

c) Support for Open and Closed-Source Models: The framework should be designed to handle both open-source and proprietary models. For closed-source systems, benchmarking will be conducted based solely on publicly accessible model outputs, ensuring that evaluations reflect what any user can achieve with the model. This approach ensures fairness and reproducibility while avoiding the need to access sensitive or proprietary details of the model's architecture or training process.

d) Dynamic Update Mechanisms: Given the evolving nature of AI models, benchmarking results must accommodate updates to model versions, retraining efforts, or new evaluation tasks. Blockchain's append-only architecture allows for tracking version histories and updating benchmarks without compromising the integrity of past records.

D. Benchmarking Process for AI Models

We propose the following benchmarking process for AI models, leveraging blockchain integration to ensure transparency, reproducibility, and fairness. The process consists of the following steps:

a) Task Selection: A diverse set of benchmark tasks is selected, covering key performance dimensions (e.g., accuracy, robustness, efficiency). These tasks should reflect real-world scenarios and may include standardized benchmarks like ImageNet [15] or GLUE [16].

b) Benchmark Selection and Definition: Benchmarking begins by identifying relevant evaluation tasks and datasets. These benchmarks must reflect the model's intended applications, covering a wide range of capabilities such as:

- Vision tasks (e.g., ImageNet [15]).
- Language tasks (e.g., GLUE [16]).
- Multi-modal tasks for models capable of handling text, images, and other data types.

Each task is paired with predefined metrics to ensure consistent evaluation. For example, accuracy, F1 scores, and latency may be chosen depending on the use case.

c) Model Evaluation: The evaluation process is conducted by running the AI model on selected benchmarks. Key considerations include:

- **Output-Driven Assessment:** For both open and closed-source models, the evaluation is performed based on publicly accessible outputs. This ensures that results are reproducible and reflect real-world usage scenarios.
- **Reproducibility:** Models are evaluated in controlled environments with publicly available datasets and task definitions to ensure consistency across evaluations.

d) Result Validation: Results are validated by independent evaluators or automated scripts to confirm adherence to the benchmarking protocols. Decentralized oracles can be used to verify performance metrics before storing them on the blockchain.

e) On-Chain Storage: The validated benchmark results, along with metadata such as model version, and evaluation metrics, are stored on the blockchain. The following components are recorded on-chain:

- Task descriptions and evaluation metrics.
- Model versions and associated model cards (e.g., minted as NFTs).
- Performance metrics and validation reports.

Storing these components on the blockchain guarantees immutability and enables stakeholders to access a complete history of a model's performance.

f) Dynamic Updates and Versioning: Updates to models (e.g., retraining or fine-tuning) require re-evaluation and version tracking. New results are appended to the blockchain, preserving the integrity of past records while reflecting improvements or changes in the model.

g) Stakeholder Access and Analysis: Benchmarking data stored on-chain is made accessible through user-friendly interfaces. Stakeholders can query and analyze data for comparisons across models or domains, leveraging aggregated insights for decision-making.

E. Blockchain-Enabled Benchmarking in Practice

By implementing benchmarking within a blockchain environment, stakeholders can establish a decentralized, transparent ecosystem for AI model evaluation. This includes, but is not limited to:

- Model cards (minted as NFTs) can link directly to on-chain benchmark results, ensuring an immutable connection between a model's documentation and its performance.
- Benchmark evaluations can use decentralized oracles to verify results and record them on-chain, ensuring consistency and integrity.
- Aggregated benchmarking data can support advanced analytics, enabling comparative studies across models and domains in a fully transparent manner.

V. Discussion

This section reflects on the requirements outlined in Sections II to IV, summarizing key insights and addressing challenges.

A. Summary

The proposed framework addresses accessibility, certification, and benchmarking as foundational pillars for integrating AI into the Cardano ecosystem. A standardized API framework (Section II) ensures accessibility through scalability and detailed documentation, enabling seamless interaction between AI models and decentralized applications. Certification (Section III) promotes transparency and accountability by using model cards to document essential model information, immutably stored on-chain. Benchmarking (Section IV) builds on this foundation, providing transparent and verifiable performance evaluations to facilitate fair comparisons across AI models.

B. Challenges

Significant challenges remain in implementing the framework. Ensuring API scalability in decentralized environments requires optimizing network protocols to handle growing demand efficiently. Defining comprehensive certification standards for model cards involves balancing the need for detailed information with simplicity and usability. Finally, storing benchmarking results on-chain requires finding efficient methods to balance transparency and storage costs while maintaining verifiability.

C. Future Work: Consensus Mechanisms

The next step is the integration of consensus mechanisms to validate AI outputs in decentralized environments. While traditional blockchain consensus algorithms (e.g., Proof of Work or Proof of Stake) ensure ledger consistency, AI-specific consensus will require innovative approaches. These mechanisms may leverage ensemble strategies, using certified and benchmarked models (Sections III and IV) to weight contributions and improve reliability. Such methods would enhance fault tolerance, mitigate the impact of malicious or faulty participants, and build trust in decentralized AI systems. The development of these mechanisms will be the focus of subsequent research milestones.

VI. Conclusion

This paper proposes a framework for integrating artificial intelligence (AI) into the Cardano blockchain, focusing on the foundational pillars: accessibility, certification, and benchmarking. By addressing key requirements and challenges, the framework establishes a scalable, transparent, and verifiable foundation for decentralized AI systems. Future work will explore consensus mechanisms to further enhance trust and reliability, building on the groundwork laid here to advance AI adoption within the Cardano ecosystem.

References

[1] R. T. Fielding, "Architectural styles and the design of network-based software architectures"; doctoral dissertation," 2000. [Online]. Available: https://ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf

[2] D. Renzel, P. Schlebusch, and R. Klamma, "Today's top "restful" services and why they are not restful," in *Web Information Systems Engineering - WISE 2012*, X. S. Wang, I. Cruz, A. Delis, and G. Huang, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 354–367.

[3] C. Rodriguez, M. Baez, F. Daniel, F. Casati, J. Trabucco, L. Canali, and G. Percannella, "Rest apis: A large-scale analysis of compliance with principles and best practices," 06 2016, pp. 21–39.

[4] T. C. Lethbridge, S. E. Sim, and J. Singer, "Studying software engineers: Data collection techniques for software field studies," *Empirical Software Engineering*, vol. 10, no. 3, pp. 311–341, 2005. [Online]. Available: https://doi.org/10.1007/s10664-005-1290-x

[5] B. A. Myers, A. J. Ko, T. D. LaToza, and Y. Yoon, "Programmers are users too: Human-centered methods for improving programming tools," *Computer*, vol. 49, no. 7, pp. 44–52, 2016.

[6] J. Bloch, "How to design a good api and why it matters," in *Proc. 21st ACM SIGPLAN Conference (OOPSLA)*, Portland, Oregon, 2006, pp. 506–507. [Online]. Available: http://portal.acm.org/citation.cfm?id=1176617.1176622

[7] L. Floridi and J. Cowls, "A Unified Framework of Five Principles for AI in Society," *Harvard Data Science Review*, vol. 1, no. 1, jul 1 2019, https://hdsr.mitpress.mit.edu/pub/l0jsh9d1.

[8] A. Jobin, M. Ienca, and E. Vayena, "The global landscape of ai ethics guidelines," *Nature Machine Intelligence*, vol. 1, pp. 389 – 399, 2019.

[9] M. Mitchell, S. Wu, A. Zaldivar, P. Barnes, L. Vasserman, B. Hutchinson, E. Spitzer, I. D. Raji, and T. Gebru, "Model cards

for model reporting," in *Proceedings of the Conference on Fairness, Accountability, and Transparency*, ser. FAT* '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 220–229. [Online]. Available: https://doi.org/10.1145/3287560.3287596

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[11] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986. [Online]. Available: https://doi.org/10.1038/323533a0

[12] 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.

[13] D. S. V. Madala, M. P. Jhanwar, and A. Chattopadhyay, "Certificate transparency using blockchain," in *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, 2018, pp. 71–80.

[14] S. Pu and J. S. L. Lam, "The benefits of blockchain for digital certificates: A multiple case study analysis," *Technology in Society*, vol. 72, p. 102176, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0160791X22003177

[15] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[16] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman, "GLUE: A multi-task benchmark and analysis platform for natural language understanding," in *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, T. Linzen, G. Chrupała, and A. Alishahi, Eds. Brussels, Belgium: Association for Computational Linguistics, Nov. 2018, pp. 353–355. [Online]. Available: https://aclanthology.org/W18-5446

[17] D. Raji, E. Denton, E. M. Bender, A. Hanna, and A. Paullada, "Ai and the everything in the whole wide world benchmark," in *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, J. Vanschoren and S. Yeung, Eds., vol. 1, 2021. [Online]. Available: https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/084b6fbb10729ed4da8c3d3f5a3ae7c9-Paper-round2.pdf

[18] P. Liang, R. Bommasani, T. Lee, D. Tsipras, D. Soylu, M. Yasunaga, Y. Zhang, D. Narayanan, Y. Wu, A. Kumar, B. Newman, B. Yuan, B. Yan, C. Zhang, C. A. Cosgrove, C. D. Manning, C. Re, D. Acosta-Navas, D. A. Hudson, E. Zelikman, E. Durmus, F. Ladhak, F. Rong, H. Ren, H. Yao, J. WANG, K. Santhanam, L. Orr, L. Zheng, M. Yuksekgonul, M. Suzgun, N. Kim, N. Guha, N. S. Chatterji, O. Khattab, P. Henderson, Q. Huang, R. A. Chi, S. M. Xie, S. Santurkar, S. Ganguli, T. Hashimoto, T. Icard, T. Zhang, V. Chaudhary, W. Wang, X. Li, Y. Mai, Y. Zhang, and Y. Koreeda, "Holistic evaluation of language models," *Transactions on Machine Learning Research*, 2023, featured Certification, Expert Certification. [Online]. Available: https://openreview.net/forum?id=iO4LZibEqW

[19] L. Gao, J. Tow, B. Abbasi, S. Biderman, S. Black, A. DiPofi, C. Foster, L. Golding, J. Hsu, A. Le Noac'h, H. Li, K. McDonell, N. Muennighoff, C. Ociepa, J. Phang, L. Reynolds, H. Schoelkopf, A. Skowron, L. Sutawika, E. Tang, A. Thite, B. Wang, K. Wang, and A. Zou, "A framework for few-shot language model evaluation," 07 2024. [Online]. Available: https://zenodo.

org/records/12608602

[20] S. Biderman, H. Schoelkopf, L. Sutawika, L. Gao, J. Tow, B. Abbasi, A. F. Aji, P. S. Ammanamanchi, S. Black, J. Clive, A. DiPofi, J. Etxaniz, B. Fattori, J. Z. Forde, C. Foster, J. Hsu, M. Jaiswal, W. Y. Lee, H. Li, C. Lovering, N. Muennighoff, E. Pavlick, J. Phang, A. Skowron, S. Tan, X. Tang, K. A. Wang, G. I. Winata, F. Yvon, and A. Zou, "Lessons from the trenches on reproducible evaluation of language models," 2024. [Online]. Available: https://arxiv.org/abs/2405.14782