# A Knowledge-Based System for Managing Hardware Dependency and Reproducibility in Quantum Machine Learning Workflows

1st Thilanka Munasinghe
*Information Technology & Web Science*
*Rensselaer Polytechnic Institute*
New York, USA
munast@rpi.edu

2nd Kimberly A. Cornell
*Department of Information Sciences and Technology*
*University at Albany*
Albany, New York, USA
0000-0001-9551-9689

3rd James Hendler
*Department of Computer Science*
*Rensselaer Polytechnic Institute*
Troy, New York, USA
hendler@cs.rpi.edu

4th George Berg
*Department of Cybersecurity*
*University at Albany*
Albany, New York, USA
gberg@albany.edu

5th Jennifer C. Wei
*NASA Goddard Space Flight Center*
*Greenbelt,*
Maryland, USA
jennifer.c.wei@nasa.gov

***Abstract*—In this position paper, we emphasize the importance of a Knowledge Graph-based system for managing hardware dependencies and reproducibility in Quantum Machine Learning (QML) workflows, especially when attempting to evaluate the quantum advantage. QML applications can benefit from integration with a Knowledge Graph (KG) to effectively organize, contextualize, and scale information for complex problem-solving using Quantum Computing (QC) techniques. QC-based Quantum Machine Learning (QML) is emerging as a field for solving complex computational problems that are challenging for classical (i.e., non-quantum) systems. However, reproducibility and benchmarking against classical machine learning (CML) models remain challenging due to the varied and evolving quantum hardware and computational techniques, as well as the intricate nature of the datasets. By leveraging KGs to recognize and abstract beyond these variables, QML can use CML approaches more generally. This can extend to handling heterogeneous, interconnected datasets, particularly in domains that require spatiotemporal and relational modeling. Using a QML application as a use-case example from environmental analytics, we demonstrate that this approach enhances interpretability, scalability, and adaptability across various QC hardware, quantum algorithms, and applications across different domains.**

***Index Terms*—Quantum Machine Learning, Knowledge Graph, Machine Learning, Quantum Hardware, Workflows**

## I. Introduction

The recent emergence of Quantum Computing (QC) as a mainstream computing method has yielded developments with significant benchmarking results from leading tech companies such as IBM and Google [1]. This has encouraged the use of Quantum-inspired machine learning (QML). Consequently, researchers are already evaluating the applicability and effectiveness of these new QML approaches on real-world problems across many fields [2]–[5]. IBM has provided a framework for articulating an operational definition of quantum advantage and for evaluating it [6]. Reproducibility and the ability to validate the QC workflow are significant factors in achieving the quantum advantage. Furthermore, a QML model's performance is not determined by its algorithm and data alone; it is critically dependent on the complex relationship between the specific quantum hardware used (e.g., superconducting, trapped-ion, or photonic) and the chosen method for encoding classical data (e.g., Amplitude, Basis, or Angle Encoding) into Quantum States (QSs) [5], [7]. This deep dependence on hardware type, encoding technique, and algorithmic choice creates significant variability, severely hindering reproducibility, cross-platform comparisons, and the generalization of solutions.

### A. Current Challenges in QML process

Unlike classical machine learning (CML), which benefits from well-established benchmarks and produces consistent results across many hardware platforms, the novel field of QC and quantum-inspired ML faces unique and foundational challenges. This is especially true for QML workflows, which are deeply coupled to the rapidly evolving, highly variable underlying quantum hardware. Performance benchmarks are not easily comparable across platforms such as superconducting qubit systems (IBM, Google), trapped-ion computers, and photonic devices. The literature shows that QML implementations critically depend on the use of different types of quantum devices [8]–[10].

Furthermore, different architectures are specialized for different computational models; for example, quantum annealers are designed for optimization problems [11], while gate-based superconducting systems are used for circuit-based algorithms [12]. This hardware-algorithm complexity is leading to a vast amount of domain knowledge and implementation information that is hard to keep track of the implementation workflows [2]–

[5], [13]–[15]. This complexity is compounded by a lack of standardized documentation for other critical workflow components. The choice of how to encode classical data into quantum states, for instance, is a critical, often hardware-dependent decision with no clear best practices as of the writing of this paper. Without a structured, transparent representation of these dependencies, the end-to-end QML workflow is opaque. This "reproducibility gap" hinders the field's growth, challenges the "verifiability" of the QML application process, makes it difficult to generalize findings, and poses a significant risk of wasted resources on pursuing non-viable experimental paths when implementing a QML application based on an example.

### B. The Case for a Knowledge-Based Approach to QML

We recommend integrating Knowledge Graphs (KGs) to model the QML workflow, effectively managing implementation complexities beyond simple documentation. By representing the workflow as a KG, we can capture essential relationships among components like experiments, datasets, hardware, algorithms, and encoding methods. This creates a structured, queryable, hardware-specific, and reproducible knowledge framework. The implementation of a KG within a published QML application's workflow shows that any QML application supported by a KG can enhance its epistemological foundation by organizing the knowledge associated with the QML process [15], [16].

## II. Background and Related Work

This section details the core challenges in QML that necessitate a knowledge-based approach and reviews existing work in QML ontologies. To support our position, we outline the following core arguments:

1) **QML Lacks Standardization and Reproducibility** Current QML applications are difficult to reproduce due to *hardware dependencies, encoding method variations, and evolving QC techniques*. Unlike classical ML, where datasets and models can be easily shared, QML experiments rely on *specific quantum devices and encoding strategies*, making direct replication challenging. KGs can *document QML workflows, encoding decisions, and hardware requirements*, enabling researchers to *trace, reproduce, and refine* past experiments.
2) **Hardware Dependency Bias in QML Performance** QML results depend on the *type of quantum hardware used (e.g., superconducting, trapped-ion, photonic), the number of qubits available, and error rates*. Benchmarks for QML algorithms are currently biased towards *specific quantum platforms*, leading to misleading conclusions about their superiority over CML. KGs can *store metadata about hardware attributes and their impact on QML models*, allowing for *transparent comparison* across different quantum platforms.
3) **Encoding Classical Data into QSs is a Bottleneck** Different *quantum encoding techniques* (e.g., *Amplitude Encoding, Basis Encoding, Angle Encoding*) affect the efficiency of the QML model. The choice of encoding method is *highly dependent on the available quantum hardware and dataset size*, influencing computational feasibility. KGs can help *catalog encoding techniques, map them to quantum hardware constraints, and link them with appropriate QML models*, providing a *decision-support system* for encoding selection.
4) **Cost Considerations in QC must be Structured** Running QML applications on quantum hardware is *expensive*, requiring careful optimization of *qubit usage, gate depth, and execution time*. The lack of a standardized cost analysis framework makes it difficult for researchers to estimate *resource efficiency* before running quantum experiments. KGs can *track cost-related parameters*, enabling researchers to *predict execution costs and optimize QML workflows accordingly*.
5) **KGs for Interoperability and Knowledge Sharing** A structured *ontology-driven approach* can organize QML concepts, ensuring interoperability across different QC ecosystems. KGs facilitate *semantic search, knowledge sharing, and automation of workflow documentation*, accelerating QML adoption in *real-world applications* such as *climate action, finance, and healthcare*. By structuring relationships between *datasets, encoding methods, algorithms, and hardware*, KGs can enable *automated recommendations for QML model selection*.

### A. The Data Encoding Bottleneck

A critical challenge in QML is encoding classical data into QSs, with techniques like `Amplitude`, `Basis`, and `Angle Encoding` varying in computational cost and efficiency. The choice of encoding depends on the specific problem, dataset size, and the capabilities of quantum hardware. As noted in our previous work [13]–[15], this choice can create bottlenecks and inconsistencies in workflows. Thus, a knowledge graph is essential for cataloging these techniques, linking them to hardware constraints, and associating them with the corresponding QML models, serving as a valuable decision-support system as expressed in Figure 2.

### B. Related Work in QML Ontologies

The need for organizing quantum information is not a new idea. Martyniuk et al. explain the need for ontologies in QC as new algorithms and hardware are developed. They introduce the "PlanQK Project," which provides a platform for knowledge transfer and vendor-agnostic access to QC resources [17]. Similarly, Martens et al. propose the QuantumShare ontology to capture and share essential knowledge to support collaboration between QC researchers [18]. While these projects provide a valuable foundation for a general QC ontology, our work differs significantly. We focus specifically on the QML application workflow, presenting the process of a queryable implementation based on a case study [13] that provides the foundation to demonstrate how a KG can be used to diagnose tangible reproducibility problems.

In the following sections, we discuss our position, which takes a different perspective from the current approach of

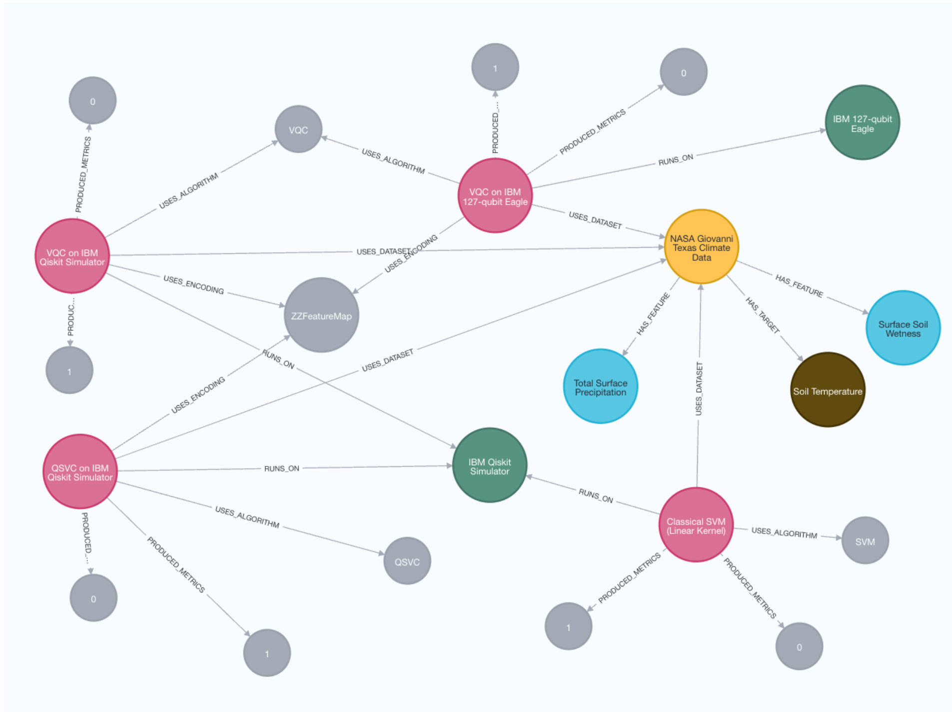


Fig. 1. KG example that capture the information of QML workflow based on our Use Case, with nodes “0” and “1” representing binary classes that lead to crop frosting or not [13].

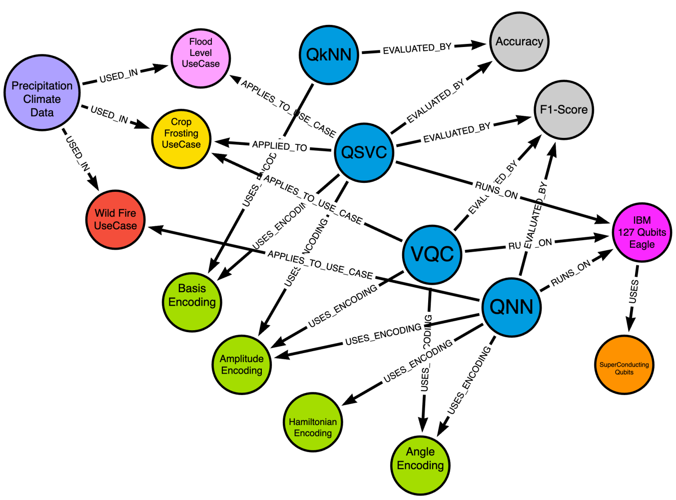


Fig. 2. Visual representation of knowledge organization showing QML models such as Quantum Support Vector Classifier (QSVC), Variational Quantum Classifier (VQC), Quantum k-Nearest Neighbor (QkNN) and data encoding techniques such as Basis Encoding, Amplitude Encoding etc, for precipitation data in climate related applications (flooding, wildfire, crop-frosting).

TABLE I
CORE RELATIONSHIP TYPES IN THE QML-KG SCHEMA

| **Relationship Type** | **Description** |
|---|---|
| `USES_ALGORITHM` | `(Experiment)` → `(Algorithm)` |
| `RUNS_ON` | `(Experiment)` → `(Hardware)` |
| `USES_DATASET` | `(Experiment)` → `(Dataset)` |
| `USES_ENCODING` | `(Experiment)` → `(Encoding)` |
| `PRODUCED_METRICS` | `(Experiment)` → `(Metrics)` |

simply implementing QML applications and documenting their workflows. Acceptable QML results on real-world applications cannot be solely comparisons with current classical ML results; the QML algorithms, the type of hardware used for QC, and the number of qubits available can produce markedly different results in QML experiments.

Our core argument is that quantum data encoding and the representation of classical data on a quantum computer (QC) are highly variable. Each step of the application workflow needs to be organized from the information organization perspective to determine why a specific quantum algorithm is being used, under what conditions the chosen data encoding techniques are valid, based on the type of algorithm being used, and the hardware that is used to conduct the QML application.

The solution we propose is the use of KGs and ontologies, which help provide a proper understanding of how to advance the development of QML applications for real-world use and improve the reproducibility of the work, clearly expressing how core-relationship types can be expressed through a QML-KG schema as an example illustrated in Table I when all information about the ML workflow is properly organized using KGs. Secondly, we argue that an end-to-end QML workflow must be clear, including assumptions about hardware, qubits, algorithms, and data encodings, before results can be meaningfully interpreted, repeated, or built upon. Knowing this before investing time and resources is key. Otherwise, QML applications might waste money and resources on futile attempts, as actual hardware based QC is very expensive due to high operational costs.

Thirdly, based on the type of QC used, hardware bias could be introduced to the QML application. The QML applications depend highly on the type of QC and the number of qubits available [5]. The architecture, error rates, and connectivity of qubits directly influence the feasibility and performance of QML algorithms [6], [7]. As QCs evolve, researchers need a systematic way to organize, share, and reproduce knowledge about how different QCs can be used for QML tasks. KGs offer an efficient, optimal solution for structuring this knowledge, enabling researchers to explore pathways tailored to specific hardware constraints, algorithmic requirements and accuracy metrics being used as illustrated in the KG example shown in Figure 1 based on the use case [13] we used.

### C. Epistemological Aspect of the Position and Arguments

Organizing a knowledge base for QC types and the number of qubits available in each device through a KG provides a structured, efficient, and reproducible framework for advancing research. By linking hardware attributes, algorithms, and applications, the KG enables researchers to make informed decisions and share their findings transparently. We believe this approach accelerates progress in QML and fosters collaboration and innovation across the QC community. Cost analysis is critical given QCs' unique requirements and limitations. Factors such as qubit usage, gate fidelity, circuit depth, execution time, and quantum hardware access fees influence costs. A KG can effectively organize and represent these aspects, offering researchers and practitioners a structured way to evaluate and compare the cost of QML applications. This ensures informed decision-making and promotes transparency in resource utilization.

A universal fault-tolerant QC that is capable of efficiently solving problems like large integer factorization and efficiently searching an unstructured database requires a very large number of qubits with very low error rates and extended coherence times, which is currently a challenging task to achieve [19], [20]. The current experimental progress towards universal fault-tolerant QCs is expected to be achieved only several decades from now [21]. However, noisy intermediate-scale quantum (NISQ) computers are already available today, making significant milestones in the field of QC [10], [22]. NISQ devices, although limited by noise and a relatively small number of qubits, provide a valuable platform for experimentation and learning [22]. They allow researchers and developers to investigate practical quantum applications, optimize quantum circuits, and identify which problems might benefit the most from specific QC types.

We support our position with (1) the organization of the QML workflow using a KG and (2) empirical observations of QML applications across various encoding techniques and QC systems, including IBM Quantum simulator [23] and applications with our published real worl QML based application as an example use case [13]. Implementing a KG for that use case illustrated in figures 1 and 2, which enhance the understanding and advancement of that developed QML application for crop-frosting prediction using binary classification using NASA Earth observational data and help the research community on improving reproducibility of the example QML use case [13] by properly organizing workflow information.

Secondly, we argue that if the end-to-end QML workflow is not clear enough to evaluate before implementation by an independent developer who wishes to adopt the work from a research paper and implement the work for their customized application, they must be able to understand the specific QML system used to avoid expensive dead ends or unexpected biases. QML applications depend highly on the type of QC and the number of qubits available [24], [25]. The architecture, error rates, and connectivity of qubits directly influence the feasibility and performance of QML algorithms [22], [26]. As QCs evolve, researchers need a systematic way to organize, share, and reproduce knowledge about how different QCs can be used for various QML tasks [22], [27].

Therefore, KGs can offer an efficient, optimal solution for structuring this knowledge, enabling researchers to explore pathways tailored to specific hardware constraints and associated algorithmic requirements [28], [29]. Using a KG to organize QML approaches based on QC types and number of qubits availabile in the QC system used (i.e 127 qubits IBM Quatum System One or 156 qubits based IBM Quantum System Two) provides a structured, efficient, and reproducible framework to advance research. By linking hardware attributes, algorithms, and applications, the KG enables researchers to make informed decisions and share their findings transparently. We believe this approach accelerates progress in QML and fosters collaboration and innovation across the QML research communities.

Cost analysis is critical given the unique requirements and limitations of QCs. Factors such as number of qubits usage (127 or 156 qubits), circuit depth, execution time, and quantum hardware access fees influence costs. A KG can effectively organize and represent these aspects, offering researchers and practitioners a structured way to evaluate and compare the cost ahead of QML application implementation. This ensures informed decision-making and promotes transparency in resource utilization and results comparisons.

QML approaches must scale to tackle large problems (e.g., those requiring 100+ qubits), and integrating High-Performance Computing (HPC) becomes essential [30]–[32]. In the past, with classical computing, HPC provided the classical computational backbone needed for preprocessing, quantum-classical hybrid workflows, and error mitigation in QC to effectively use NISQ systems is an emerging approach [6]. However, managing the complexities of QML+HPC workflows for large-scale problems requires an efficient organizational framework [6], [32], [33]. Therefore, using KGs to facilitate project workflows by integrating metadata, dependencies, and workflows across QML and HPC systems will be critical. This could potential improve efficiency, ensures reproducibility, and supports informed decision-making.

## III. Conclusion

QML is a promising field, but it faces severe reproducibility and standardization challenges. To illustrate these challenges, our research, previously published as a paper [13], uses a real-world case study that applies QML classifiers (VQC and QSVC) to a NASA Earth Observation dataset for climate related problem. In that work, we discovered that the same VQC algorithm yielded different performance when run on a simulator versus a real 127-qubit IBM QC. This experience provides concrete evidence for the urgent need for a better information and knowledge organizational framework. Each step of the QML application workflow needs to be organized from the information organization perspective, including a KG framework to capture, organize, and query these complex QML workflows for managing complexities of hardware-algorithm dependency and enhance the reproducibility of QML applications and experiments.

## Acknowledgment

We thank the Future of Computing Institute at RPI for the 127-qubit IBM QC used in this study [13]. We also appreciate the support from the University at Albany. Additionally, thanks to NASA's Goddard Space Flight Center and GES DISC for their assistance in obtaining EO satellite data.

## References


[1] D. A. Abanin, R. Acharya, L. Aghababaie-Beni, G. Aigeldinger, A. Ajoy, R. Alcaraz, I. Aleiner, T. I. Andersen, M. Ansmann, F. Arute *et al.*, "Constructive interference at the edge of quantum ergodic dynamics," *arXiv preprint arXiv:2506.10191*, 2025.

[2] J. Chow, O. Dial, and J. Gambetta, "Ibm quantum breaks the 100-qubit processor barrier," *IBM Research Blog*, vol. 2, 2021.

[3] J.-G. Liu, Y.-H. Zhang, Y. Wan, and L. Wang, "Variational quantum eigensolver with fewer qubits," *Physical Review Research*, vol. 1, no. 2, p. 023025, 2019.

[4] J. Wang, G. Guo, and Z. Shan, "Sok: Benchmarking the performance of a quantum computer," *Entropy*, vol. 24, no. 10, p. 1467, 2022.

[5] M. Schuld, I. Sinayskiy, and F. Petruccione, "An introduction to quantum machine learning," *Contemporary Physics*, vol. 56, no. 2, pp. 172–185, 2015.

[6] O. Lanes, M. Beji, A. D. Corcoles, C. Dalyac, J. M. Gambetta, L. Henriet, A. Javadi-Abhari, A. Kandala, A. Mezzacapo, C. Porter *et al.*, "A framework for quantum advantage," *arXiv preprint arXiv:2506.20658*, 2025.

[7] M. Schuld and N. Killoran, "Quantum machine learning in feature hilbert spaces," *Physical review letters*, vol. 122, no. 4, p. 040504, 2019.

[8] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, "Parameterized quantum circuits as machine learning models," *Quantum Science and Technology*, vol. 4, no. 4, p. 043001, 2019.

[9] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio *et al.*, "Variational quantum algorithms," *Nature Reviews Physics*, vol. 3, no. 9, pp. 625–644, 2021.

[10] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke *et al.*, "Noisy intermediate-scale quantum algorithms," *Reviews of Modern Physics*, vol. 94, no. 1, p. 015004, 2022.

[11] I. Hen and F. M. Spedalieri, "Quantum annealing for constrained optimization," *Physical Review Applied*, vol. 5, no. 3, p. 034007, 2016.

[12] S. A. Wilkinson and M. J. Hartmann, "Superconducting quantum many-body circuits for quantum simulation and computing," *Applied Physics Letters*, vol. 116, no. 23, 2020.

[13] T. Munasinghe, P. Lai, J. Wei, J. Hendler, and K. A. Cornell, "Assessment of quantum ml applicability for climate actions: Comparison of the variational quantum classifier and the quantum support vector classifier with classical ml models," in *2024 IEEE International Conference on Big Data (BigData)*. IEEE, 2024, pp. 4357–4366.

[14] T. Munasinghe, "Utilization of Classical and Quantum Machine Learning-Based Models to Study the Human Migration Dynamics," Ph.D. Dissertation, University at Albany, State University of New York, 2025, electronic Theses & Dissertations (2024 - present), No. 186. [Online]. Available: https://scholarsarchive.library.albany.edu/etd/186

[15] T. Munasinghe, K. A. Cornell, J. C. Wei, G. Berg, and J. Hendler, "A knowledge graph framework for organizing heterogeneous datasets for utilization in classical and quantum computing: Current challenges and future directions," in *2024 IEEE International Conference on Big Data (BigData)*. IEEE, 2024, pp. 8781–8785.

[16] P. R. Giri, M. Kurokawa, and K. Saito, "Quantum negative sampling strategy for knowledge graph embedding with variational circuit," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 2. IEEE, 2023, pp. 280–281.

[17] D. Martyniuk, M. Falkenthal, N. Karam, A. Paschke, and K. Wild, "An analysis of ontological entities to represent knowledge on quantum computing algorithms and implementations." in *Qurator*, 2021.

[18] J. Martens, I. Kumara, G. Monsieur, W.-J. V. D. Heuvel, and D. A. Tamburri, "Quantumshare: Towards an ontology for bridging the quantum divide," in *International Conference on Conceptual Modeling*. Springer, 2023, pp. 412–429.

[19] D. Bacon, J. Kempe, D. A. Lidar, and K. B. Whaley, "Universal fault-tolerant quantum computation on decoherence-free subspaces," *Physical Review Letters*, vol. 85, no. 8, p. 1758, 2000.

[20] M. Grassl, B. Langenberg, M. Roetteler, and R. Steinwandt, "Applying grover's algorithm to aes: quantum resource estimates," in *International Workshop on Post-Quantum Cryptography*. Springer, 2016, pp. 29–43.

[21] Q. A. Memon, M. Al Ahmad, and M. Pecht, "Quantum computing: navigating the future of computation, challenges, and technological breakthroughs," *Quantum Reports*, vol. 6, no. 4, pp. 627–663, 2024.

[22] E. Gil-Fuster, J. Eisert, and C. Bravo-Prieto, "Understanding quantum machine learning also requires rethinking generalization," *Nature Communications*, vol. 15, no. 1, p. 2277, 2024.

[23] P. Rao, K. Yu, H. Lim, D. Jin, and D. Choi, "Quantum amplitude estimation algorithms on ibm quantum devices," in *Quantum Communications and Quantum Imaging XVIII*, vol. 11507. SPIE, 2020, pp. 49–60.

[24] M. Schuld and N. Killoran, "Is quantum advantage the right goal for quantum machine learning?" *Prx Quantum*, vol. 3, no. 3, p. 030101, 2022.

[25] J. D. Martín-Guerrero and L. Lamata, "Quantum machine learning: A tutorial," *Neurocomputing*, vol. 470, pp. 457–461, 2022.

[26] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven, and J. R. McClean, "Power of data in quantum machine learning," *Nature communications*, vol. 12, no. 1, p. 2631, 2021.

[27] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, 2017.

[28] Y. Ma and V. Tresp, "Quantum machine learning algorithm for knowledge graphs," *ACM Transactions on Quantum Computing*, vol. 2, no. 3, pp. 1–28, 2021.

[29] Y. Ma, V. Tresp, L. Zhao, and Y. Wang, "Variational quantum circuit model for knowledge graph embedding," *Advanced Quantum Technologies*, vol. 2, no. 7-8, p. 1800078, 2019.

[30] M. Schulz, M. Ruefenacht, D. Kranzlmüller, and L. B. Schulz, "Accelerating hpc with quantum computing: It is a software challenge too," *Computing in Science & Engineering*, vol. 24, no. 4, pp. 60–64, 2022.

[31] S. T. Bieberich and M. A. Sandoval, "Analyzing machine learning performance in a hybrid quantum computing and hpc environment," *arXiv preprint arXiv:2407.07294*, 2024.

[32] K.-C. Chen, X. Li, X. Xu, Y.-Y. Wang, and C.-Y. Liu, "Quantum-classical-quantum workflow in quantum-hpc middleware with gpu acceleration," in *2024 International Conference on Quantum Communications, Networking, and Computing (QCNC)*. IEEE, 2024, pp. 304–311.

[33] M. Riedel, G. Cavallaro, and J. A. Benediktsson, "Practice and experience in using parallel and scalable machine learning in remote sensing from hpc over cloud to quantum computing," in *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*. IEEE, 2021, pp. 1571–1574.