WJAETS
World Journal of Advanced Engineering Technology and Sciences
eISSN: 2582-8266
Cross Ref DOI: 10.30574/wjaets
Journal homepage: https://wjaets.com/

(REVIEW ARTICLE)

GRAPHICAL USER INTERFACE FOR VISUALIZING AND MANAGING CLOUD-BASED DATA PIPELINES

Raghu Gopa *

Wilmington University, 320 N Dupont Hwy, New Castle, Delaware 19720, USA.
* Corresponding Author

World Journal of Advanced Engineering Technology and Sciences, 2026, 20(02), 209–215

Publication history: Received on 10 June 2026; revised on 18 August 2026; accepted on 20 August 2026

Article DOI: https://doi.org/10.30574/wjaets.2026.20.2.0340

ABSTRACT

Cloud computing and big data technologies have greatly increased the use of cloud-based data pipelines for the ingestion, transformation, and analysis of large and heterogeneous datasets. The growing complexity of these pipelines, however, creates substantial challenges for their design, implementation, monitoring, and maintenance, particularly for users with limited programming expertise. Graphical User Interfaces (GUIs) have been proposed as a solution that reveal in a straightforward way pictorial ecosystems that communicate with data pipelines. GUIs allow practitioners to configure complicated workflow settings without low-level code through direct visual interaction. This review describes GUI-based administration which would model and operate cloud-based data pipelines. The existing methodologies were systematically reviewed within the paper through which conceptual underpinnings, as well as implementation plans, had been outlined clearly. The survey takes into account the fundamental principles of cloud pipeline architecture, data flow modalities and the principles of GUI design and examines how the combination of these principles contributes to the level of usability, scalability and operational efficiency. Key design principles such as modularity, drag-and-drop functionality, and real-time state visualization are discussed. This discussion then assesses the functional activity of GUIs in the workflow construction, real time monitoring and data visualization and concomitantly assesses the aspects of the system that meets these characteristics, and user-contextual dashboards. The paper has also evaluated tooling and platforms as examples of pipeline governance using the GUI and compared their functionality, performance indicators, and limitations.

Keywords: Cloud Computing, Data Pipelines, Graphical User Interface, Workflow Visualization, Distributed Systems

1 INTRODUCTION

The rapid growth of cloud computing and big-data technologies has transformed various fields. Modern applications rely on data pipelines that organize a series of steps, including data ingestion, transformation, processing, and visualization. These pipelines enable the execution of data processing functions to be scalable and distributable such that organizations can effectively handle large volumes of data. However, controlling and tracking these pipelines pose significant challenges especially to professionals who are not proficient in advanced programming.

Historically, command-line interfaces and scripting languages have been used as the basis of the design and implementation of data pipelines but can be technically demanding. This dependency creates a significant barrier for domain experts that must concentrate on the analytical objectives rather than the challenge of deploying the system.

Graphical user interfaces (GUIs) have become effective tools for facilitating data pipeline design, implementation and monitoring. GUIs provide interactive and graphical processes through which a user may contribute to complicated systems via drag-and-drop components of user interfaces, visual processes and interactive switches. Empirical research indicates that GUI-based pipeline systems contribute greatly to usability by allowing users to construct and run workflows without extensive programming knowledge [1], [2]. Simultaneously, using graphical workflow systems as an example, one can offer the visual representation of the process of connecting processing steps which seem distant one to another, which simplifies the pipeline architecture perception and eases the process of introducing changes to it. In addition to that, the systems enhance reproducibility and transparency with the provision of concise illustrations of data streams and processing involved in the form of graphics [2]. This is particularly important in large-scale cloud environments where multiple users collaborate and where the users share computing resources. Other strengths of GUI-based cloud pipeline platforms include scalability, flexibility, and remote accessibility as well as operating on cloud computing and have GUIs. With these systems, users can design and implement workflow on distributed systems without directly managing the underlying hardware [3]. Visualization tools that are a part of the GUIs are also useful in real-time monitoring and performance indicators and therefore enable users to identify the bottlenecks and effectively optimize the workflows [4]. Even with the above pros, there are still a number of undeveloped concerns in the creation of effective GUI systems that would be adjusted to cloud-based pipelines, specifically, in the context of scalability, utility, and system performance. It implies that a general overview of the existing methodologies, architectures and tools is actually highly warranted to make sense of current trends and, in the same vein, be in a position to determine the possible directions of future research. The current paper provides an extensive review of how graphical user interfaces are used to visualize and manage data pipelines on the cloud based on its structure and design, features, most popular tools, and future trends.

2 BACKGROUND AND CONCEPTS

Cloud-based data pipelines have become a significant component of current data-driven systems, which attain automated data moving and changes of information in a distributed environment. These pipelines typically consist of multi-stage processes that include data ingestion, preprocessing, analysis, and storage that are organized by cloud infrastructures. The pipelines can scale capacity up or down in response to workload demands and share the similar aim of computer resources hence, as the physical infrastructure is no longer bound with the dependency on it and the customer base can readily make use of tremendous amounts of data across the geographically dispersed systems successfully. Moreover, cloud pipelines can be configured with many data sources and predictiveness, and hence, they can be applicable in the multi-domain, complicated applications [5]. At the same time, the graphical user interfaces have become important tools that have enhanced ease of inter-operating with complicated computing architectures. A GUI will give people an abstract representation of what is happening behind the scenes, and therefore the ability to easily use systems by interacting with them through graphical representation such as nodes, connectors and dashboards rather than explaining the code itself. GUIs can also be used in data pipelines to allow the user to visualize the processes in a pipeline and create visual links between processing units that are unrelated with one another and also lower the technical barrier that is often associated with the construction of the pipeline. This diagrammatic approach to methodology enhances the mental models of the users of the data flow, as well as the system behavior, that is described by big and heterogeneous datasets [6] in the above sets. Another feature that adds to the richness of usability is the combination of the GUIs with cloud pipelines as it enables users to monitor and control the distributed processes in real time. Interactive dashboards allow users to observe the execution, check the intermediate outputs, and edit workflow dynamically. These capabilities are necessary to offer transparency and reproducibility especially when there is collaborative research and enterprise environments. At the same time, GUI-based systems are frequently accompanied by metadata management, and a visualization feature, which is why users can navigate the composition of relations within data, accordingly, being able to extract insights, more effectively [7]. The other defining trait of the GUI-based pipeline systems is the fact that they support the modular and reusable workflow elements. Such systems would foster standardization and interoperability through encapsulation into discrete modules. This modular structure not only simplifies the pipeline construction process but works towards the process of sharing to other users and organizations. Therefore, cloud pipelines are beneficial in improving productivity, collaboration, and processing projects [8].

3 ROLE OF GUI IN DATA PIPELINE MANAGEMENT

Graphical user interfaces (GUIs) have become an essential component of cloud-based data-pipeline operations by translating complex tasks into graphical interactions. In the conventional systems, the process of writing scripts and manually defining the dependencies is performed and can be error-prone and time-consuming. In comparison, GUI-based systems allow end users to construct pipelines through graphic constituent connection of an illustration of modular square blocks and thus allows even end-users with little skill to assemble elaborate procedures. This abstraction of the images will tremendously improve usability and reduce the learning curve of pipeline development thereby making the interdisciplinary data processing more accessible and easier to learn [9]. Another significant use of GUIs with regard to data-pipeline management is linked to real-time monitoring and control of execution. To observe the condition of each stage of the pipeline, the modern GUI systems provide dashboards that allow displaying the rate of execution, the amount of resources used, and the potential failures. By the interactive quality of these monitoring features, one is able to immediately identify the presence of bottlenecks and failures, and under other conditions as well, the capability of restarting the workflow at any given point without rerunning the entire pipeline. They are used to enhance operational efficiency as well as reduce the cost of calculation particularly in large-scale cloud deployment that is characterized by shared and dynamically billed resources [10].

Visualization is a component that is also heavily addressed by GUI in the management of data pipelines. By providing streams of data, intermediate outputs and the results of analysis in graphical format, GUI can help the users to better understand how complex systems behave. Such visualizations are beneficial in terms of debugging and optimization and also helpful in terms of communication within the team as they understand the logic of pipelines very well. Additionally, built-in reporting instruments in GUIs allow the consideration of performance indicators and resource usage and improve the decision-making and optimization of the system [11]. Also, pipeline systems using GUI allow standardization and automation by packaging workflows into templates. Users can store, share, and replicate pipelines in various environments, and thereby obtain consistency and reproducibility in data-processing activities. This is particularly beneficial in cloud-based environments, where teamwork and distributed computing are the norm. The ability to combine the cloud services, containerization technologies, and workflow engines in a single graphical user interface facilitates the simplified coordination of the intricate pipelines and maintains the flexibility and scalability [12]. To further illustrate the capabilities of various GUI-based pipeline systems, a comparison of representative tools is given below (Table 1).

Table 1: Comparison of GUI-Based Cloud Data Pipeline Tools

Tool	GUI Type	Cloud Support	Key Features	Limitations
CloVR	Web-based GUI	Yes	Automated pipeline execution, cloud integration	Limited customization
STORMSeq	Web GUI	Yes	User-friendly cloud execution, low cost	Limited advanced controls
Nextpie	Web dashboard	Yes	Resource monitoring, visualization reports	Focused on reporting only
RAP Pipeline	Web GUI	Yes	Modular workflow, graphical outputs	Domain-specific (RNA-Seq)

4 DEMAND SIGNAL ENGINEERING

Demand Signal Engineering (DSE) is a new paradigm in data-driven systems, where data-driven responses are focused on the capture, analytical processing, and responsive action regarding real-time data signals originating in distributed systems. In the framework of cloud-based data pipelines, DSE highlights the manipulation of the raw data streams into actionable information through the continuous processing, adaptive workflows, and intelligent coordination. In contrast to conventional pipelines, which are batch oriented, demand-driven systems are dynamic systems, responding to the incoming patterns of data and the requests of users in real time. Such a transition requires new pipeline architectures, which can be used effectively to handle high-velocity data and still maintain accuracy and consistency through sequential processing. Graphical user interfaces play a central role in the process of demand signal engineering by providing user-friendly resources to the development and administration of adaptive workflows.

Users can configure event-driven triggers, conditional logic and also data routing applications without much programming knowledge. This is especially relevant when it comes to a context in which rapid decisions are required, such as healthcare analytics, financial systems, and large-scale scientific research. GUI systems enable users to view the incoming data stream as well as visualize trends and dynamically alter the behavior of a pipeline, which makes them real time responsive and enable them to optimize the level of operational efficiency [13]. The other dimension of demand signal engineering that can be highlighted is the combination of components of workflow in the form of scalable cloud infrastructure. Computational advantage afforded by cloud environments is needed in responding to fluctuations in data scales, and flexible architecture of pipelines it enables these needs to change in a more modular manner. This is enhanced by GUI based systems that allow one to construct and reconstruct workflows with reusable components in such a manner that pipelines can be reconfigured for new demand signals. This also supports interoperability between different tools and platforms, a common flow of data and developmental synergy among the different tools is also made possible by this portability [14].

The visualization and analytics component are one of the key elements in DSE whereby the solution allows users to make inferences and identify trends in complex data. Graphical feedback on data flow, system performance and analysis results in real-time delivered through GUI based dashboards enable users to identify trends, anomalies, and potential system inefficiencies. These visual tools in addition to aiding in the augmentation of the situational awareness assist in the data-driven decision-making process by ensuring that the information is provided in a manner that is simple to access and in an interactive manner. Moreover, contemporary GUI systems are also endowed with reporting and tracking capabilities to ensure that the users can study previous data and streamline the pipeline performance over a long period of time [15].

Finally, demand signal engineering needs to have resilient mechanisms that ensure reliability, reproducibility and consumption of resources. These requirements can be fulfilled by pipeline systems based on the GUI and provide features such as workflow versioning, automatic processing of errors, and monitoring resources. The features enable the user to be constant in performance but adapt to dynamic data situations. Along with this, packaged integration of containerization and cloud orchestration technologies in GUI environments gives the capacity to implement and dynamically span pipelines to numerous computational infrastructures through which the total system can be more resilient and adjustable with changes in demand and resource requirements [16] (Figure 1).

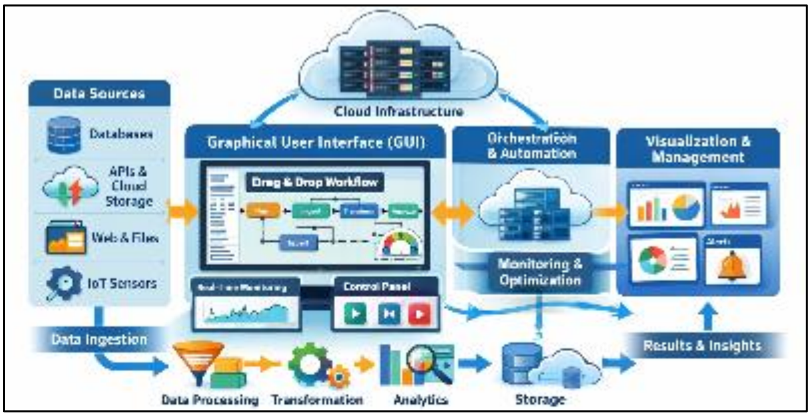

Figure 1: Overview of GUI-Based Cloud Data Pipeline Architecture

5 EXISTING TOOLS AND SYSTEMS

The development of graphical user interfaces for cloud-based data pipelines has led to numerous tools and platforms which have been customized to support workflow modeling, implementation, and monitoring. These tools seek to reduce the workload and complexity of big-data processing through the synthesis of the visualization, automation and cloud-computing systems within one environment. Among the salient trends of these tools is the emphasis on accessibility which enables users with little knowledge of programming to create and execute intricate data pipelines through the assistance of graphical user interfaces. Modern pipeline systems based on GUI often use workflow management engines that support scalable and modular pipeline design. These systems allow users to define complex workflows that are interconnected components and that can be deployed on distributed cloud infrastructures. As an example, integrated pipeline environments have demonstrated the ability to support end-to-end data processing, that is, combining several analytical tools into a single graphical interface. These platforms not only accelerate workflow development but also improve interoperability of the different computational tools and data formats, making them highly adaptable to a variety of applications [17].

Another highly important type of tools is focused on the provisioning of cloud-native environments through a seamless integration of data processing, storage, and visualization capabilities. These platforms enable the users to directly instantiate pipelines on cloud environments and still maintain overall control with graphical dashboards. They allow users to focus on the analytical work instead of configuration of the underlying infrastructure by removing the underlying infrastructure complexities. Moreover, these systems typically involve the automation of resources, workflow scheduling, and real-time monitoring features that have a profound positive impact on optimizing the functioning and minimizing manual errors [18]. Along with full-scale pipeline platforms, it is possible to state that certain tools are designed to improve a specific aspect of pipeline management, such as reporting and performance analysis. The tools offer graphical dashboards that integrate the execution metrics, resource utilization data and workflow logs into a single interface and, therefore, can give users an insight into the performance of pipelines, identify the areas of inefficiency, and enable the process of resource allocation to become optimized. Such reporting systems are especially useful in cloud environments, where the efficient utilization of resources directly influences the cost and performance [19]. One may see that the existing GUI-based systems can be considered as a significant step toward more unified, scalable, and user-friendly systems. Despite the differences in the characteristics and the purpose of use of each of the mentioned platforms, they all highlight the importance of visualization, automation, and integration into clouds in the modern data pipelines management. Such systems will provide a foundation around which the subsequent developments will be centered on the enhancement of usability, scaling and intelligent automation of the cloud-based data processing environments (Figure 2).

Figure 2: Key Features of GUI for Cloud-Based Data Pipelines

6 CHALLENGES AND LIMITATIONS

User interfaces are of great benefit to cloud-based pipelines management, nevertheless, a number of challenges and constraints remain, and complicates further utilization and successful operation of these interfaces. First is the problem of scalability especially when big data streams of high velocity are being encountered. Although it is true that cloud-based infrastructures in a way offer elastic resources, there are issues associated with GUI-based systems in making adequate representations of and controlling the complexities of highly complex processes that have many tightly coupled components. As a pipeline becomes more complex, the display becomes increasingly cluttered making it less useful and therefore complicating the process of the user to interpret and control the system [20]. The second significant limitation is performance overhead and resource management. Such systems have the propensity of introducing additional abstraction between the user and the underlying computation processes. Although these abstractions facilitate interaction during interaction, in reality, they could be a source of inefficiencies during implementation particularly when dealing with real-time applications or applications that require substantial resources. In cloud-computing environments where services are billed based on usage, ineffective pipeline implementation can cause higher operation costs. Moreover, it can also be difficult to align the graphical interface with backend processes especially on a distributed system that will assist in serving a great number of simultaneous users [21]. As they tend to operate on the same cloud resources and may be utilized by multiple users, who may operate with sensitive information, the necessity to secure the data transfer, the robust access control, the adherence to the regulatory standards will become imperative. To guarantee integrity and confidentiality of data, the GUI platforms are therefore expected to have strong authentication systems, role-based access-control systems as well as encryption systems. Such safeguards, however, would be rather complex to the system designers to accomplish without affecting usability [20]. Additionally, limitations in flexibility and customization are also evident. Even though the systems based on GUI are built to ensure that the task of designing a workflow is simplified, it might restrict more advanced users who need the capability to make more fine adaptations to the setups of pipelines. Hard-coded modules and graphical components could restrict the usage of extremely specific or non-standard workflows. This is a sustained issue of GUI-based design of pipeline systems, as opposed to usability and flexibility. As well, the lack of uniformity among the tools and technologies being incorporated in a single interface is likely to result in compatibility issues particularly in competing with the rapidly expanding cloud services and data-processing models [22]. Overall, even though the degree of accessibility and ease of use is significantly advanced, the problems of scalability, performance, security and flexibility need to be addressed to secure further evolution and popularity of the GUI-based cloud pipeline systems.

7 FUTURE DIRECTIONS

It can be expected that the continued maturation of the graphical user interfaces (GUIs) in the data pipelines based on the cloud is marked by significant achievements in the field of artificial intelligence, automation, and flexible system design. The biggest opportunity is the integration of AI-augmented capabilities into the GUI-based environment which would enable intelligent workflow recommendations, error detection and self-optimization in pipelines. With the help of such systems, it is possible to learn about the past trends of implementation and even suggest an alteration of the pipeline layout to reduce the amount of manual interventions and enhance the overall productivity. With new developments of the AI technologies, the GUI-based platforms will gain more proactive and context-responsive characteristics, thus, helping users to make informed decisions with minimal intervention [23]. Another notable trend in the development of more interactive and immersive visualization techniques. Future GUI systems are expected to incorporate advanced visual analytics, and they, in turn will possess real-time data exploration, real-time dashboards, and, possibly, augmented or virtual reality visualizations of complex pipeline activities. The innovations would enable the users to have a better perception of data relations and system behaviors particularly on large and multidimensional data. More significant crucial roles will be played by better visualization schemes in enabling transparency and interpretability to data driven systems [24]. Moreover, the increasing use of hybrid and multi-cloud environments will create the need to be more adaptable and interoperable GUI-based pipeline infrastructures. The next set of platforms will be those that require the seamless integration of the services of other cloud vendors and the experience of their customers to stay the same. This will require interface standardization, and increased API connectivity and process portability. The GUI systems will also increase in concentration in enabling the users to write pipelines to be executed using various infrastructures with minimal reconfiguration which will promote scalability and reliability in the distributed environment [25].

8 CONCLUSION

Graphical user interfaces have become essential components of cloud-based pipeline design and management, which significantly increases ease of access, user friendliness and efficiency. In most cases, GUIs enable users to develop, monitor and simplify sophisticated workflows without requiring extensive programming skills since they have friendly graphic representations. The aggregation of such capabilities is also boosted by the application of cloud computing where volumes of data can be shared in regards to the delivery of scaled infrastructures as well as accommodating infrastructures. This review has highlighted the key concepts, architectures, tools and the challenges that are related to the GUI based pipeline systems and future trends that take the said tool towards its development. Despite the significant advances achieved by modern systems, the issue of scalability, performance, and security remain challenges that require continued research and development. These are the considerations that will be made and more developed technologies including artificial intelligence and improved visualization applications will also be looked at to continue generating data pipelines systems. GUI-based cloud data pipelines will represent a giant leap in the current form of data processing to bridge the gap between complex computing requirements with easy-to-use interface, and, as a result, would constitute the key to more effective and smarter data-intensive data-driven applications.

STATEMENTS ON COMPLIANCE WITH ETHICAL STANDARDS

Disclosure of conflict of interest

The author declares no conflict of interest regarding this manuscript.

REFERENCES

[1] Kucukural, A., Garber, M., Yukselen, O., Turkyilmaz, O., Ozturk, A., Girard, I., & Martin, R. (2019). DolphinNext: A graphical user interface for creating, deploying and executing Nextflow pipelines. *bioRxiv*.

[2] Dinov, I., Torri, F., Macciardi, F., Petrosyan, P., Liu, Z., Zamanyan, A., ... Toga, A. (2011). Applications of the pipeline environment for visual informatics and genomics computations. *BMC Bioinformatics*, 12, 304.

[3] Alarcon, M. L., Oruche, R., Pandey, A., & Calyam, P. (2022). Cloud-based data pipeline orchestration platform for COVID-19 evidence-based analytics. *Novel AI and Data Science Advancements for Sustainability in the Era of COVID-19*, 159–180.

[4] Kim, B., Ali, T., Dong, C., Lijeron, C., Mazumder, R., Wultsch, C., & Krampis, K. (2019). miCloud: A plug-n-play, extensible, on-premises bioinformatics cloud for seamless execution of complex next-generation sequencing data analysis pipelines. *Journal of Computational Biology*, 26, 280–284.

[5] Nagasaki, H., Mochizuki, T., Kodama, Y., Saruhashi, S., Morizaki, S., Sugawara, H., ... Nakamura, Y. (2013). DDBJ Read Annotation Pipeline: A cloud computing-based pipeline for high-throughput analysis of next-generation sequencing data. *DNA Research*, 20, 383–390.

[6] Zhu, X., Wolfgruber, T. K., Tasato, A., Arisdakessian, C., & Garmire, L. (2017). Granatum: A graphical single-cell RNA-Seq analysis pipeline for genomics scientists. *Genome Medicine*, 9.

[7] Robertson, C., Harris, J., Wagner, B., Granger, D., Browne, K., Tatem, B., ... Frank, D. (2013). Explicet: Graphical user interface software for metadata-driven management, analysis and visualization of microbiome data. *Bioinformatics*, 29(23), 3100–3101.

[8] Brittain, J., Tsui, J., Inward, R., Gutierrez, B., Mwanyika, G. O., Tegally, H., … Kraemer, M. (2025). GRAPEVNE: Graphical analytical pipeline development environment for infectious diseases. *Wellcome Open Research*, 10.

[9] Agrawal, S., Arze, C., Adkins, R. S., Crabtree, J., Riley, D. R., Vangala, M., … Fricke, W. F. (2017). CloVR-Comparative: Automated, cloud-enabled comparative microbial genome sequence analysis pipeline. *BMC Genomics*, 18.

[10] Karczewski, K., Fernald, G., Martin, A. R., Snyder, M., Tatonetti, N., & Dudley, J. (2014). STORMSeq: An open-source, user-friendly pipeline for processing personal genomics data in the cloud. *PLoS ONE*, 9.

[11] Ghimire, B., Booth, N., Lönnberg, T., & Aittokallio, T. (2025). Nextpie: A web-based reporting tool and database for reproducible Nextflow pipelines. *Bioinformatics Advances*, 5.

[12] D'Antonio, M., D'Onorio de Meo, P., Pallocca, M., Picardi, E., D'Erchia, A., Calogero, R., … Pesole, G. (2015). RAP: RNA-Seq analysis pipeline, a new cloud-based NGS web application. *BMC Genomics*, 16(S6), S3.

[13] Bais, P., Namburi, S., Gatti, D., Zhang, X., & Chuang, J. H. (2017). CloudNeo: A cloud pipeline for identifying patient-specific tumor neoantigens. *Bioinformatics*, 33, 3110–3112.

[14] Xu, Y., Yang-Turner, F., Volk, D., & Crook, D. (2020). NanoSPC: A scalable, portable, cloud compatible viral nanopore metagenomic data processing pipeline. *Nucleic Acids Research*, 48, W366–W371.

[15] Ren, Y., Yu, G., Shi, C., Liu, L., Guo, Q., Han, C., … Huang, H. (2022). Majorbio Cloud: A one-stop, comprehensive bioinformatic platform for multiomics analyses. *iMeta*, 1.

[16] Kim, B., Ali, T., Lijeron, C., Afgan, E., & Krampis, K. (2017). Bio-Docklets: Virtualization containers for single-step execution of NGS pipelines. *GigaScience*, 6, 1–7.

[17] Cianfrocco, M., Lahiri, I., DiMaio, F., & Leschziner, A. (2018). An integrated, pipeline-based approach for cryo-EM structure determination and atomic model refinement in the cloud. *bioRxiv*.

[18] Ren, Y., Yu, G., Shi, C., Liu, L., Guo, Q., Han, C., … Huang, H. (2022). Majorbio Cloud: A one-stop, comprehensive bioinformatic platform for multiomics analyses. *iMeta*, 1.

[19] Ghimire, B., Booth, N., Lönnberg, T., & Aittokallio, T. (2025). Nextpie: A web-based reporting tool and database for reproducible Nextflow pipelines. *Bioinformatics Advances*, 5.

[20] Lemus Alarcon, M., Oruche, R., Pandey, A., & Calyam, P. (2022). Cloud-based data pipeline orchestration platform for COVID-19 evidence-based analytics. *Novel AI and Data Science Advancements for Sustainability in the Era of COVID-19*, 159–180.

[21] Torka, S., & Albayrak, S. (2024). ALPACA: Adaptive learning pipeline for comprehensive AI. *arXiv*.

[22] Dinov, I., Lozev, K., Petrosyan, P., Liu, Z., Eggert, P. R., Pierce, J., … Toga, A. (2010). Neuroimaging study designs, computational analyses and data provenance using the LONI pipeline. *PLoS ONE*, 5.

[23] Torka, S., & Albayrak, S. (2024). ALPACA: Adaptive learning pipeline for comprehensive AI. *arXiv*.

[24] Robertson, C., Harris, J., Wagner, B., Granger, D., Browne, K., Tatem, B., … Frank, D. (2013). Explicet: Graphical user interface software for metadata-driven management, analysis and visualization of microbiome data. *Bioinformatics*, 29(23), 3100–3101.

[25] Xu, Y., Yang-Turner, F., Volk, D., & Crook, D. (2020). NanoSPC: A scalable, portable, cloud compatible viral nanopore metagenomic data processing pipeline. *Nucleic Acids Research*, 48, W366–W371.