Libraries, Integrations and Hubs for Decentralized AI using IPFS

Richard Blythman[1] Mohamed Arshath[1] Jakub Smékal[1]
Hithesh Shaji[1] Sal Vivona[1] Tyrone Dunmore[1]

[1] Algovera.ai

Abstract

AI requires heavy amounts of storage and compute. As a result, AI developers are regular users of centralised cloud services such as AWS, GCP and Azure, compute environments such as Jupyter and Colab notebooks, and AI Hubs such as HuggingFace and ActiveLoop. There services are associated with certain benefits and limitations that stem from the underlying infrastructure and governance systems with which they are built. These limitations include high costs, lack of monetization and reward, lack of control and difficulty of reproducibility. At the same time, there are few libraries that allow data scientists to interact with decentralised storage in the language that data scientists are used to, and few hubs where they can discover and interact with AI assets. In this report, we explore the potential of decentralized technologies - such as Web3 wallets, peer-to-peer marketplaces, decentralized storage (IPFS and Filecoin) and compute, and DAOs - to address some of the above limitations. We showcase some of the libraries and integrations that we have built to tackle these issues, as well as a proof of concept of a decentralized AI Hub app, that all use IPFS as a core infrastructural component.

Keywords: AI Hubs; Data Marketplaces; Decentralized AI; IPFS; Web3

1 Introduction

The field of deep learning is powered by assets such as datasets, models and software, which require a powerful underlying infrastructure consisting of components like storage and compute. Schwartz et al. (2020) discuss how the expense associated with the 300,000x increase in compute requirements from 2012 to 2018 raises the barrier for participation in AI research (while also being environmentally unfriendly). Recently, there has been a trend towards open source software in AI, which has made significant contributions to research and applications (Langenkamp and Yue (2022)). For example, the majority of models are implemented in open source libraries such as TensorFlow and PyTorch developed in the open source language Python.

An AI Hub is a platform that allows data scientists, engineers and other stakeholders to discover, share and collaborate on AI assets such as code, datasets, models, apps, notebooks, pipelines and other software. AI Hubs have made significant contributions to the democratization of state-of-the-art research. For example, source code is commonly made available through hubs such as GitHub and HuggingFace Hub. Other assets such as datasets and pre-trained model weights are often open sourced through hubs such as Kaggle, HuggingFace and ActiveLoop Hub. As a result, data scientists are regular users of AI Hubs to provide a place for assets to be stored, shared and further developed.

Despite much progress, many assets remain inaccessible to data scientists outside large organization. Datasets like JFT-300M (Sun et al. (2017)) by Google and models like GPT-3 (Brown et al. (2020)) remain siloed for competitive reasons, while other companies and universities may lack the technical know-how to share assets. Furthermore, open assets can be scattered across many apps and websites, and requires the user to follow a lengthy tutorial for download, setup and processing. The assets are often isolated from compute environments and not well integrated with workflows, making reproducibility difficult.

While the assets themselves may be open source, the platform itself is almost always closed source and governed by a centralized entity. The trade-offs that have been made may arise from the availability and maturity of technologies and governance structures with which they are built. Today's AI Hubs tend to rely exclusively on centralised cloud services such as AWS, GCP, and Azure, and the high expense of these services is often passed on to the user. Cloud infrastructure is a critical component for training large scale models, and the computational cost for re-training or fine-tuning of open source models has become prohibitive and created a barrier-to-entry. Furthermore, the platform ultimately controls the accessibility of uploaded assets, and act as gatekeepers to which assets are allowed to exist on the respective platform. Finally, the platform can also monetize the network effects of user contributions without sharing in the rewards (Kumar et al. (2020)).

Data science collaboration is complex and involves many stakeholders and stages (Zhang et al. (2020)). Model training is just one step within a pipeline, that also includes a number of other steps that make up the data science workflow. Many workflows are closed source with large companies building their own proprietary data pipelines (Uvarov et al. (2022)). Langenkamp and Yue (2022) expect that data-centric tooling is the next frontier for AI research, although they note that the incentives for open sourcing these tools - such as competitive differentiation - are different. Machine learning tooling for the rest of the pipeline is fragmented and suffers a tragedy of the commons.

Decentralized technologies such as peer-to-peer storage, compute and marketplaces, machine learning frameworks and decentralized autonomous organizations (DAOs) present opportunities for tackling the above challenges. We explore the benefits offered by these technologies to address some of the above issues. We propose a combination of a number of these technologies to offer an alternative value proposition to existing solutions. Metahub[1] is a decentralized, permissionless and censorship-resistant platform for data scientists, engineers, domain experts and users to build AI systems. Algovera is a community that is coming together to abide by certain standards when creating and working with digital objects and building decentralized AI applications. The community agrees to recognize each others property rights according to a certain set of rules, and achieve consensus on ethical, safety, and alignment considerations. This arrangement can ultimately be coded in software, with collective governance to update the standards. This structure of coordination facilitates a pluralist approach to AI (Siddarth (2021)).

In Section 2, we review existing AI Hubs. Based on our findings, we discuss some of the problems with existing hubs and associated libraries in Section 3. Then, in Section 4, we discuss the potential advantages of decentralized technologies for these services. Finally in Section 5, we present some of the hubs, libraries and frameworks that we have integrated with IPFS.

2 Existing AI Hubs and Libraries

The forms of existing AI Hubs have evolved as the requirements of AI and ML have become more clear. In this section, we review some existing AI Hubs such as GitHub, HuggingFace (HF) and ActiveLoop in terms of features and user base (as summarized in Table 1).

[1] https://metahub.algovera.ai/

Table 1: Existing AI Hubs

Existing AI Hub	GitHub	Huggingface	ActiveLoop	Replit
Launch	2008	2016	2018	2016
Users	SWEs	Data Scientists (NLP)	Data Scientists (CV)	SWEs
IDE	No	No	No	Yes
Payments	No	No	No	No
Storage/Asset	Code	Code, Datasets, Weights	Datasets	Code
Compute/Hosting	No	GPU (Inference)	No	CPU
Cloud Infrastructure	Centralized	Centralized	Centralized	Centralized
Governance	Centralized	Centralized	Centralized	Centralized

2.1 GitHub

GitHub[2] is a platform for software development and version control that was launched in 2008. It builds on the open source git standard and adds a number of components for efficient collaboration, CI/CD, debugging, and unit testing. It also provides social networking functionality such as feeds, followers, and wikis. The site allows users to browse public repositories for software assets on the site. As of 2019, it had more than 40 million users and 100 million repositories (GitHub (2019)).

GitHub has been heavily used for the development of AI and ML software in the form of tools, frameworks, and libraries. The number of AI and ML repositories increased slowly from 2009-2012 until an acceleration in 2017 (Gonzalez et al. (2020)). More applications of AI and ML are created annually than tools, libraries, and frameworks. It provides storage for software assets, but not large files such as datasets and model weights. The importance of these assets are an important differentiator between traditional software development and AI. To modify a code asset, a user employs a separate interactive code editor (IDE) and then pushes to GitHub through git at the command line.

2.2 HuggingFace Hub

HuggingFace (HF) is a company aiming to becoming the "GitHub for AI". HF achieved success by creating a unified library for the popular transformer models for natural language processing (NLP), making it easier to train, optimize, and deploy state-of-the-art model

[2] https://github.com/

Wolf et al. (2019). They subsequently launched HF Hub[3], a platform with features for code-sharing and collaboration such as discussions and pull requests (similar to GitHub). Unlike GitHub, HF also provide storage for large files (through git-lfs) such as datasets and pre-trained model weights (as well as code), meaning that ML developers can keep all of their assets in one place. Furthermore, users can use HuggingFace Spaces to host web-based demos of machine learning apps using the Gradio or Streamlit. They have built a large community of active data scientists, with the platform recently passing 100,000 users and 50,000 open source ML models.

To update an asset on HuggingFace Hub, data scientists must use a separate interactive code editor (IDE) with git through the command line.

2.3 ActiveLoop Hub

Activeloop Hub[4] is an open source library for efficiently storing and retrieving large machine learning datasets. Its data is stored in a columnar database management system, which is used to efficiently relate different files to each other to generate data samples via indexing. The core of the data layout is ActiveLoop's chunking mechanism, which splits a dataset into so called chunks of data of size 16 MB, accelerating data streaming by sending more data in a single network request. The data layout allows efficient grouping of different parts of the dataset which is then automatically read by other services within the ActiveLoop ecosystem, such as the data visualizer.

2.4 Replit

Replit[5] is an online IDE. Unlike GitHub and HuggingFace, where modifying assets requires a separate IDE and command line, Replit users can interact with code and source control for their project through a web-based graphical user interface. Replit provides a shared compute engine that provides collaborative coding similar to Google Docs, where code can be run and displayed to multiple users. However, GPU support has not yet been released. Furthermore, file storage is limited to 0.5 GB for free users and 5 GB for paid users, which

[3] https://huggingface.co/

[4] https://www.activeloop.ai/

[5] https://replit.com/

is too small for most ML assets. Replit has other features such as AI-assisted tools for software development, such as co-pilot and live chat and in-line threads for discussions around code by users.

3 Problems with Existing AI Hubs and Libraries

In the previous section, we explored the features of existing AI Hubs. With this information, we now analyze some of the issues with existing solutions.

3.1 High Storage and Compute Costs

The field of deep learning requires heavy amounts of storage and compute. Machine learning datasets often reach into the 100s of GBs, and pre-trained model weights can be large too. Furthermore, the computational cost of AI research is increasing exponentially with time, creating to higher barriers to entry for participants (Schwartz et al. (2020)). As a result, cloud services, such as storage and compute, are a significant expense for AI startups. Currently, three companies make up approximately two thirds of the market share of cloud service (WPOven (2022)). More than half of Amazon's profits has come from Amazon Web Services (AWS), and 20% of AWS customers deliver 80% of revenue with the widest margins come from small and medium-sized customers (CNBC (2021)). Popular AI Hubs like GitHub, HuggingFace, ActiveLoop and Replit rely exclusively on centralised cloud platforms.

3.2 Lack of Monetization and Reward

There are few online platforms where data scientists can get paid to work independently (Kumar et al. (2020)). GitHub Sponsors allows users to make monthly money donations to projects hosted on GitHub, but contributions are typically low. HuggingFace, ActiveLoop and Replit do not enable monetization by users. The payment infrastructure is primarily set up to transfer payments from the user to the platform itself. For example, while HuggingFace do offer free services and contribute to open source development, they also charge users for premium services that are not open source. In contrast, all contributions by users must be for free, with no ability to offer paid services. Data scientists often contribute to open source but still need to support themselves with income from elsewhere, such as jobs within tech companies or universities.

Open source tools and libraries are widely used by commercial platforms and products within software development and AI (Langenkamp and Yue (2022)), although the contributions are not typically rewarded. Platforms invite assets to be uploaded by users, but do not share any generated revenue or platform ownership with users, even when directly monetizing their contributions. For example, GitHub Copilot is a commercial product for code generation that uses a model trained on user-contributed code. HuggingFace's paid inference API can be used to accelerate the deployment of user-contributed models.

3.3 Lack of Control and Ownership

Generally, software developers and data scientists do not have full control and autonomy with their creations on centralized platforms. In one case, GitHub reverted malicious changes (and suspended the account) of a developer to their own popular open source library, raising questions around the rights of developers to do what they wish with their code **?**. In the field of AI, there has been an ongoing discussion on whether open sourcing disruptive models should be commonplace, since there is the potential for harm and bias. For example, AlphaFold can be used for discovery of novel toxic molecules. Language models can be trained on abusive content and used by online bots. Large models that are trained on the corpus of internet data reproduce bias within generated text and images. As a result, platforms like HuggingFace have come under pressure to gate or remove access to models. On the other hand, it can be argued that open sourcing the model puts the technology in the hands of more people that can study and solve issues around safety and bias. In other words, there is an orthogonal risk involved with centralization of AI in the hands of a few. Keeping models closed source effectively turns large tech companies into gatekeepers, who may not always be relied upon to adjudicate on disputes in an unbiased manner.

Finally, it is difficult for owners to manage fine-grained access to assets. It is common for data scientists to need to register for access to datasets online. After making a request to the owners (sometimes with information on the intended use case), the data scientist receives a link or a login. This could use a traditional access token like OAuth 2.0 (Hardt

(2012)) or API keys. This link or login for datasets and models can be widely shared, and licenses (e.g. restricting to non-commercial use) are often broken. While possible to keep repositories private, this is often a paid feature and the encryption key is held by the platform rather than the user.

3.4 Difficult to Reproduce

The limitations of existing hubs such as GitHub for AI can make reproduciblity more difficult. For example, academic papers usually contain links that may include code on GitHub, and datasets and model weights stored on the cloud. Reproducing experiments is difficult and can require many steps such as downloading datasets, running processing scripts and installing environments, which is a time-consuming and tedious user experience. This issue results from a variety of factors such as the lack of standardisation and interoperability of in the format of assets (such as dataset and code), and the decoupling of assets from compute environments and infrastructure needed to operate on them. Some of these issues can be resolved by using containers and notebooks to replicate environments and bring compute to code. At the same time, notebooks can be difficult to deploy. HuggingFace Hub uses Gradio and Streamlit apps. Replit integrates code respotiories with compute environments, but has limited storage for assets such as datasets and model weights.

3.5 Lack of Standards

Ideally, assets in an AI Hub should be modular, reusable and interoperable. The ”interface” for the modules should have a common standard. However, this typically does not happen in practice. Datasets and model weights for deep learning are distributed across many websites and cloud platforms. There are few standards for the format in which a particular type of dataset is stored e.g. file structure, file names, file types. This makes joint training of models on multiple datasets more difficult. Furthermore, models themselves often have different formats. The success of HuggingFace is largely due to the success of the transformers library, which standardized the format for this popular family of models.

4 Review of Decentralized Technologies for AI Hubs and Libraries

In Section 2, we discussed some of the features of existing AI Hubs. Decentralized technologies - such as Web3 payments, wallets, marketplaces, storage and compute, learning frameworks and DAOs - have the potential to alleviate some of the limitations of existing AI Hubs discussed above. Examples of projects working on these individual projects are shown in Figure 1.

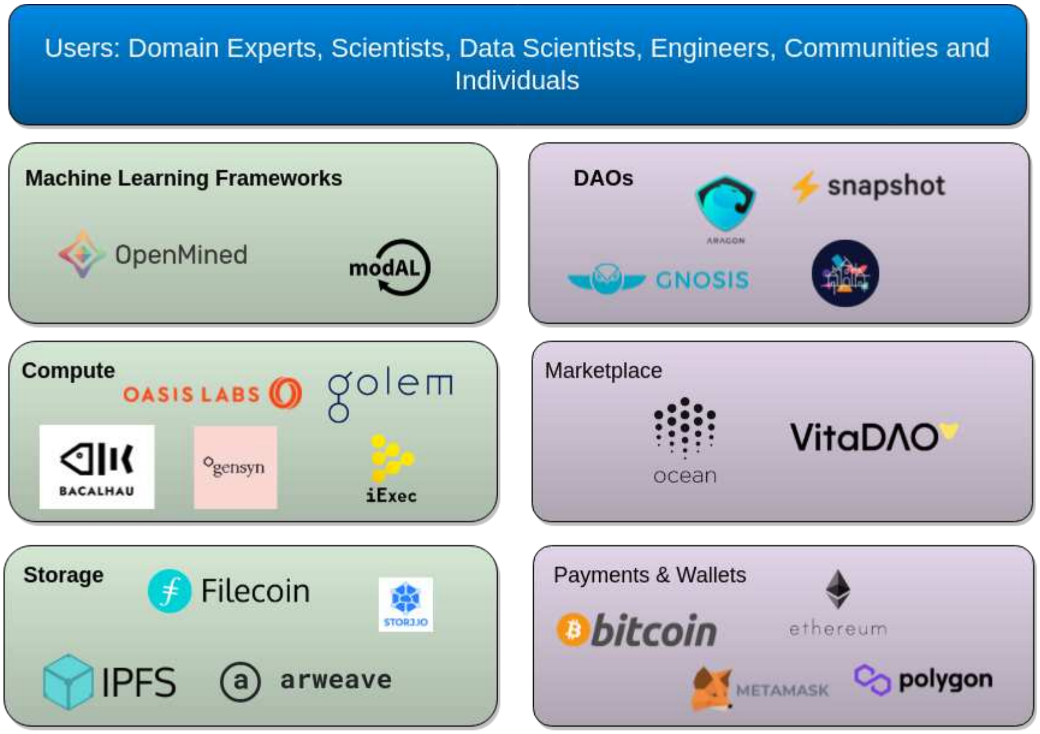


Figure 1: Decentralized infrastructure for AI Hubs, such as Web3 payments, wallets, marketplaces, storage and compute, learning frameworks and DAOs. The users of decentralized AI Hubs are the many stakeholders required for undertake successful projects.

4.1 Payments

There are few options for AI workers to monetize their creations and rewards generated by their contributions are often not shared, as discussed in Section 3.2. We believe that building in mechanisms for monetization and ownership by users would create a healthier and more sustainable ecosystem and economy. This can be achieved using cryptocurrencies (such as Bitcoin, Ethereum, Polygon, Ocean and Filecoin) and stablecoins (such as DAI or USDC), which can be used for micro- and streaming payments to stakeholders such as data scientists, data providers and compute providers with low transaction fees. Thus, integrated payments offer many opportunities for use with machine learning frameworks such as active learning and data crowdsourcing.

4.2 Web3 Wallets

As discussed in Section 3.3, data scientists typically do not have control of what they create online. The platform is typically trusted as a middle man in control of your assets. Even if a repository containing assets is private, the platform holds the private keys. A Web3 wallet can be used to put the user in control of their private keys. The word wallet tends to have financial connotations. However, wallets are often used in the real world as a place where you hold ownership and identity documents (such as a driver's license). Similarly, Web3 wallets can be used for ownership and identity in the digital world. Wallets are interoperable in the sense that you can use the same wallet to signify ownership of assets across many different protocols. It can also be used to replace a login. Web3 wallets include software wallets (such as MetaMask[6]) and hardware wallets (such as Trezor[7]).

4.3 Marketplace

Traditional AI Hubs and marketplaces are typically operated by a centralized entity serving as a middle man. In the ideal scenario, the operator provides services in exchange for transaction fees and acts as a mediator for conflict resolution between users. However, centralized hubs and marketplaces also have the power to capture an outsized proportion of the value generated in a market economy as network effects grow. Furthermore, it is difficult to manage access to assets, and licenses for datasets and software are often broken. This contributes to the issues discussed in Sections 3.2 and 3.3.

Using decentralized marketplaces protocols for tracking publication, ownership of (and access to) assets has the potential to mitigate these risks. All operations are stored on

[6] https://metamask.io/

[7] https://trezor.io/

an immutable public distributed ledger such that provenance can be tracked. For AI use cases, assets can include datasets, models, algorithms, apps, notebooks and manuscripts. Examples of decentralized marketplaces include Ocean Protocol (McConaghy (2021)) and VitaDAO (Golato and Kohlhaas (Golato and Kohlhaas)). These protocols use non-fungible tokens (NFTs) to represent ownership of the underlying intellectual property (IP), and fungible tokens to represent access rights to assets under different types of licenses. The details of published assets (and associated metadata) are encrypted and stored on-chain, along with access control parameters. A decentralized identifier (DID) is issued to represent the asset's decentralized digital identity, and a DID Document (DDO) is used to include additional information relevant to the asset. This allows providers to include information relevant to the asset, and include additional fields for accommodating any asset that may need extra fields in its description.

Access gated by tokens on a blockchain has advantages compared to traditional access token like OAuth 2.0, by solving the "double spend problem". They act as access tokens that can only be used by one individual or for a period of time. If a user receives a token on a blockchain, the user can still share it with someone else but this means the original user will no longer have access. This facilitates more fine-grained access control by owners.

4.4 Storage

While details about the assets are stored on-chain with decentralized marketplaces, the data associated with the asset are often too large to store on chain. As discussed in Section 3.1, storage on centralized cloud providers is expensive. Furthermore, these services are less robust and more prone to censorship (see Section 3.3). Popular dataset and model hubs like HuggingFace and ActiveLoop Hub rely on centralised cloud platforms.

Decentralised protocols for storage have the potential to vastly reduce the costs incurred by data scientists for storing raw and processed versions of datasets and model weights. This makes it possible to download files from multiple locations that aren't managed by a single organization. The interplanetary file system (IPFS) (Benet (2014)) is a peer-to-peer protocol for storing and accessing data in a permissionless and censorship-resistant way. IPFS clusters enable data orchestration across swarms of IPFS peers by allocating, replicating, and tracking assets. Another important feature that IPFS offers is the ability to verify the validity of assets using Content Addressable Identifiers (CIDs), based on the content's cryptographic hash. This helps both builders and the consumers of data science products. A user can verify the model used is what was promised using the inbuilt cheksum method that the CID offer.

4.5 Compute

Access to compute is a necessity for AI projects, and the provision of services by a handful of centralized companies has resulted in inflated costs (see Section 3.1). At the same time, the experiments and results of AI studies are often difficult to reproduce, as discussed in Section 3.4. While less mature than peer-to-peer storage solutions, decentralized protocols for providing compute resources aim to reduce the barrier-to-entry for compute providers and remove the centralised overheads on scaling (Fielding and Grieve (Fielding and Grieve)). This provides more options for end consumers, resulting in reduced cost. Ideally, compute should be run where the data is stored - called Compute over Data (CoD) by the Bacalhau project[8], or Compute to Data (C2D) by Ocean Protocol - rather than transporting data to the location of the compute which is expensive. In this setting, decentralized compute infrastructure presents many opportunities for integration with privacy-preserving machine learning. With Ocean Protocol, users can act as compute providers. Compute providers may run a number of compute environments that may be specialized for different use cases and stages of development such as experimenting, training and deployment.

4.6 Machine Learning Frameworks

Decentralizing infrastructure for storage and compute, and integrating payments has the potential to open up new use cases of AI. This require advancements in decentralized frameworks for machine learning. For example, privacy-preserving machine learning (PPML) - through libraries such as Openmined[9] - has the potential to unlock learning on private data such as health records and user data. Integrated payments can be used with active learning frameworks with libraries (such as modAL, Danka and Horvath (2018)) and tools for crowdsourcing human intelligence (such as Turkit, Little et al. (2010)).

[8] https://github.com/filecoin-project/bacalhau

[9] https://www.openmined.org/

4.7 DAOs

Decentralized autonomous organizations (DAOs) are systems that allow communities to coordinate and take part in self-governance, as determined by a set of self-executing rules on a blockchain (Hassan and De Filippi (2021)). DAOs have previously been suggested as governance structures for digital data trusts (Nabben (2021a)). DAOs are sometimes imagined as being governed by autonomous algorithms, with humans at the margins. However, there is an increasing push towards a future of collective intelligence that promotes harmony between humans and algorithms by optimizing for the autonomy of individuals (Nabben (2021b)). Currently, most DAOs are superficially decentralized, with most involving high concentration of voting power by a select few.

We suggest that DAOs can be used to (i) govern assets within AI Hubs, and (ii) to create decentralized AI Hubs that are governed by communities rather than single entities. Tools for governing assets within DAOs include multisignature wallets (such as Gnosis[10]) and profit-sharing mechanisms (such as Superfluid[11]). Multisignature wallets provide functionality for sharing ownership and control of assets with multiple individuals in teams in a trustless manner, while profit-sharing mechanisms can be used to distribute the revenue generated by assets. Tools for governing the infrastructure of AI Hubs include decentralized voting systems (such as Snapshot[12]).

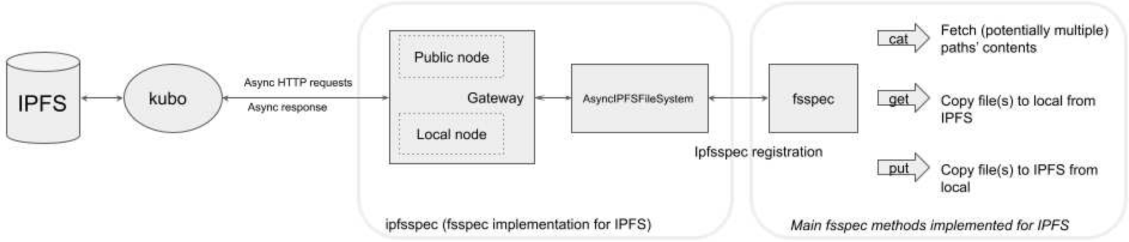
5 Integrations of IPFS with AI Hubs, Libraries and Frameworks

AI Hubs typically contain many features, as discussed in Section 2. Reproducing this functionality using decentralized technologies (as discussed in the previous section) is a large undertaking. In this section, we discuss the implemented solutions to date in the form of Python libraries, a web app and notebook environment.

[10] https://gnosis-safe.io/

[11] https://www.superfluid.finance/

[12] https://snapshot.org/


IPFS
kubo
Async HTTP requests
Async response
Public node
Gateway
Local node
AsyncIPFSFileSystem
fsspec
Ipfsspec registration
ipfsspec (fsspec implementation for IPFS)
cat
Fetch (potentially multiple) paths' contents
get
Copy file(s) to local from IPFS
put
Copy file(s) to IPFS from local
Main fsspec methods implemented for IPFS

NOTE:

- *To work with local node, a IPFS demon must be running (either IPFS Desktop, companion, or through the CLI)*
- *`put` only works on local node*

Figure 2: Schematic of ipfsspec.

5.1 Python Libraries for IPFS

Decentralised storage solutions have the potential to vastly reduce the costs incurred by data scientists for storing raw and processed versions of datasets, as well as model weights. Currently, there are few tools for interacting with decentralised storage in the language that data scientists are used to i.e. Python. This is especially true for writing to storage. Furthermore, IPFS is not well integrated with AI Hubs and libraries.

5.1.1 IPFSSpec (and HuggingFace)

Filesystem Spec (fsspec) is a project to provide a unified pythonic interface to local, remote and embedded file systems and bytes storage. It is used by HuggingFace, Pandas, and Dask. Previously, a solution for a read-only version of fsspec (called ipfsspec) existed. We implemented write functionality for ipfsspec[13], and re-implemented much of the read functionality. By doing so, read and write to IPFS is now supported by dependent libraries such as HuggingFace, Pandas, and Dask. The architecture of ipfsspec is shown in Figure 2.

5.1.2 IPFSPy

While ipfsspec has the advantage of being used by other data science libraries, it supports a limited set of functions such as cp, rm, cat and mkdir. The ipfspy[14] library provides a thin

[13] https://github.com/AlgoveraAI/ipfsspec

[14] https://github.com/AlgoveraAI/ipfspy

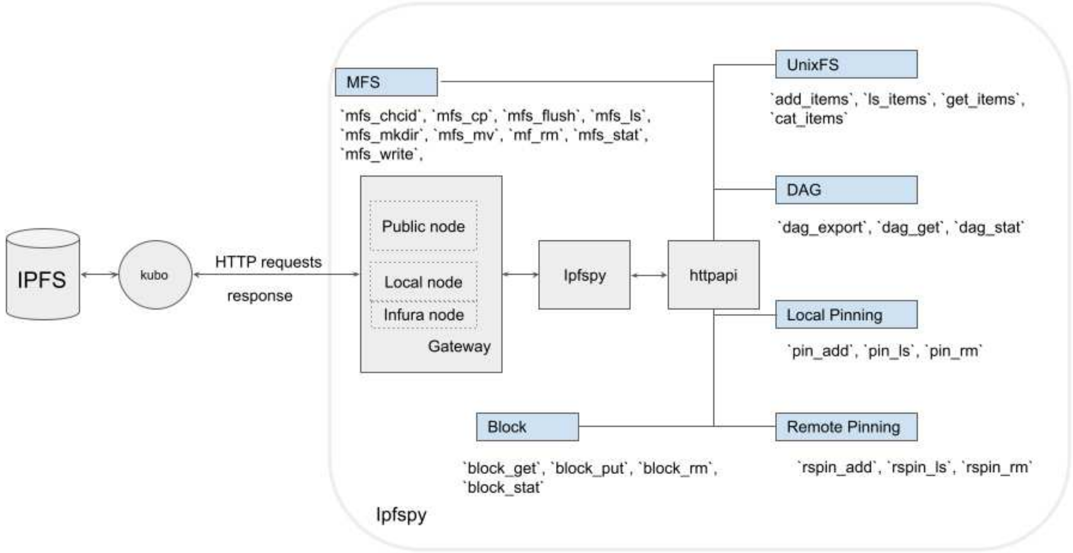

MFS
`mfs_chcid`, `mfs_cp`, `mfs_flush`, `mfs_ls`,
`mfs_mkdir`, `mfs_mv`, `mf_rm`, `mfs_stat`,
`mfs_write`,
UnixFS
`add_items`, `ls_items`, `get_items`,
`cat_items`
DAG
`dag_export`, `dag_get`, `dag_stat`
IPFS
kubo
HTTP requests
response
Public node
Local node
Infura node
Gateway
Ipfspy
httpapi
Local Pinning
`pin_add`, `pin_ls`, `pin_rm`
Block
`block_get`, `block_put`, `block_rm`,
`block_stat`
Remote Pinning
`rspin_add`, `rspin_ls`, `rspin_rm`
Ipfspy

Figure 3: Schematic of ipfspy.

wrapper around endpoints for the IPFS HTTP API, as well as Estuary and Pinata APIs. It incorporates the ipfsspec library while also providing the Python community with a wider set of functionalities for interacting with the various components of IPFS and Filecoin, such as MFS, UnixFS, DAG, Local Pinning, Remote Pinning, and Block uses. This makes it easier to build custom solutions on top of IPLD for data storage and loading within specific AI and ML use cases. At the same time, we think that re-implementing the functionality of the various building blocks of IPFS in Python (i.e. without using the IPFS HTTP API) would further improve customizability. The architecture of ipfspy is shown in Figure 3.

5.2 IPFS on ActiveLoop

Activeloop Hub is a repository for machine learning datasets and a library for efficient data streaming. The Hub introduces a data chunking mechanism extending the Zarr standard to be able to train machine learning models faster without requiring the data scientist to download large datasets locally. To enable the use of data streaming for the Web3 data science community, we integrated ipfpy within Activeloop Hub[15] to enable decentralized

[15] https://github.com/AlgoveraAI/Hub

storage for ActiveLoop Hub datasets. The integration is fully interoperable with the existing Hub library and allows the user to select any IPFS Gateway supporting read/write functionality.

5.3 IPFS on Ocean Marketplace

The Ocean Marketplace[16] is an open source community marketplace for data, built on Ocean Protocol. It allows data providers to publish and monetize data assets. Most users store their assets on centralized storage such as Google Drive. As part of core tech grants for Ocean, we integrated storage using IPFS and Filecoin.

5.4 IPFS on Metahub

There are several existing Web2 AI Hubs for datasets, models, apps and other assets, such as HuggingFace Hub and ActiveLoop Hub. However, these platforms are centralized with limitations discussed in Section 2. IPFS and Filecoin allow users to store and retrieve assets using a peer-to-peer, rather than client-server, model. However, the discoverability of assets stored on IPFS is an issue. The Ocean Protocol marketplace facilitates ownership, monetization and discoverability of assets, although it has not been tightly integrated with storage solutions. Metahub[17] is our implementation of an AI Hub that combines the best parts of IPFS and Ocean Protocol and integrates with HuggingFace and ActiveLoop Hub, to create a Web3 AI marketplace where data scientists can use datasets and generate revenue from the algorithms that they develop. We also wrote scripts for scraping HuggingFace and ActiveLoop Hub datasets, downloading them, uploading to IPFS and publishing to the marketplace.

5.5 IPFS on Streamlit

Streamlit[18] is an open source library for building interactive AI apps in Python. Its main use case is allowing data scientists to quickly build and share demos of their ML models

[16] https://market.oceanprotocol.com/

[17] https://metahub.algovera.ai/

[18] https://streamlit.io/

and it has been used extensively in open source AI communities. Streamlit is also integrated into HuggingFace Hub. However, Streamlit did not previously have support for Web3 functionality. We have integrated the Streamlit app framework with MetaMask[19], IPFS/Filecoin and Ocean Protocol, meaning that data science apps can now build on these Web3 components. For example, users can now interact with components (like buttons) to log in with their Web3 wallet, store assets on IPFS and run compute-over-data directly from Streamlit apps. In future, we plan to embed Web3-integrated Streamlit apps directly into Metahub.

5.6 IPFS in Custom Notebook Environments

Notebooks are one of the primary environments where data scientists perform exploratory data analysis (EDA) and build proof-of-concept workflows. The ipfsspec and ipfspy libaries can be used within notebook environments for interfacing with decentralized storage. In addition, we have created Jupyter Lab extensions[20] to log in with Metamask, upload data to IPFS/Filecoin and publish assets to the Ocean marketplace through the frontend.

6 Conclusion

In this report, we (i) reviewed existing hubs and libraries for AI development, (ii) discussed some of the problems with existing solutions, (iii) discussed the potential advantages of decentralized technologies for these services, and finally (iv) presented some of the hubs, libraries and frameworks that we have integrated with IPFS.

Acknowledgements

The authors acknowledge the generous support from the Filecoin Foundation (devgrant #517) and Ocean Protocol Foundation.

References

Benet, J. (2014). IPFS - content addressed, versioned, p2p file system.

[19] https://github.com/AlgoveraAI/streamlit-metamask

[20] https://github.com/AlgoveraAI/jupyterlab_extensions

Brown, T., B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems 33*, 1877–1901.

CNBC (2021). How amazon's cloud business generates billions in profit. `https://www.cnbc.com/2021/09/05/how-amazon-web-services-makes-money-estimated-margins-by-service.html`. Accessed: 2022-08-30.

Danka, T. and P. Horvath (2018). modal: A modular active learning framework for python. *arXiv preprint arXiv:1805.00979*.

Fielding, B. and H. Grieve. Gensyn: The hyperscale, cost-efficient compute protocol for the world's deep learning models.

GitHub (2019). The state of the octoverse. `https://octoverse.github.com/`. Accessed: 2022-10-18.

Golato, T. and P. Kohlhaas. Vitadao.

Gonzalez, D., T. Zimmermann, and N. Nagappan (2020). The state of the ml-universe: 10 years of artificial intelligence & machine learning software development on github. In *Proceedings of the 17th International Conference on Mining Software Repositories*, pp. 431–442.

Hardt, D. (2012). The oauth 2.0 authorization framework. Technical report.

Hassan, S. and P. De Filippi (2021). Decentralized autonomous organization. *Internet Policy Review 10*(2), 1–10.

Kumar, A., B. Finley, T. Braud, S. Tarkoma, and P. Hui (2020). Marketplace for AI models.

Langenkamp, M. and D. N. Yue (2022). How open source machine learning software shapes ai. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 385–395.

Little, G., L. B. Chilton, M. Goldman, and R. C. Miller (2010). Turkit: human computation algorithms on mechanical turk. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, pp. 57–66.

McConaghy, T. (2021). Tools for the web3 data economy. *Ocean Protocol Foundation Ltd. URL: https://oceanprotocol. com/tech-whitepaper. pdf [accessed 2020-03-09]*.

Nabben, K. (2021a). Decentralised autonomous organisations (daos) as data trusts: A general-purpose data governance framework for decentralised data ownership, storage, and utilisation. *Available at SSRN*.

Nabben, K. (2021b). Imagining human-machine futures: blockchain-based "decentralized autonomous organizations". *Available at SSRN*.

Schwartz, R., J. Dodge, N. A. Smith, and O. Etzioni (2020). Green ai. *Communications of the ACM 63*(12), 54–63.

Siddarth, D. t. (2021). How ai fails us. *Technology & Democracy Discussion Paper*.

Sun, C., A. Shrivastava, S. Singh, and A. Gupta (2017). Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pp. 843–852.

Uvarov, T., B. Tripathi, and E. Fainstain (2022, January 4). Data pipeline and deep learning system for autonomous driving. US Patent 11,215,999.

Wolf, T., L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al. (2019). Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

WPOven (2022). Cloud market share 2022: An overview of growing ecosphere. `https://www.wpoven.com/blog/cloud-market-share/`. Accessed: 2022-08-30.

Zhang, A. X., M. Muller, and D. Wang (2020). How do data science workers collaborate? roles, workflows, and tools. *Proceedings of the ACM on Human-Computer Interaction 4*(CSCW1), 1–23.