NFT-based Data Provenance for AI Transparency in Enterprise Information Systems

Verginadis, Yiannis; Almpanoudis, Orestis; Apostolou, Dimitris; Mentzas, Gregoris; Tuler de Oliveira, M.

DOI
10.1016/j.procs.2025.02.153

Publication date
2025

Document Version
Final published version

Published in
Procedia Computer Science

Citation (APA)
Verginadis, Y., Almpanoudis, O., Apostolou, D., Mentzas, G., & Tuler de Oliveira, M. (2025). NFT-based Data Provenance for AI Transparency in Enterprise Information Systems. *Procedia Computer Science*, *256*, 565-572. https://doi.org/10.1016/j.procs.2025.02.153

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 256 (2025) 565–572

Procedia
Computer Science

www.elsevier.com/locate/procedia

CENTERIS – International Conference on ENTERprise Information Systems / ProjMAN – International Conference on Project MANagement / HCist – International Conference on Health and Social Care Information Systems and Technologies 2024

NFT-based Data Provenance for AI Transparency in Enterprise Information Systems

Yiannis Verginadis[a,b,*], Orestis Almpanoudis [b], Dimitris Apostolou [c], Marcela T. de Oliveira [d], Gregoris Mentzas [b],

[a]Department of Business Administration, Athens University of Economics and Business,Patission 76, 10434 Athens, Greece
[b]Institute of Communications and Computer Systems, Iroon Polytechniou 9, 15780 Zografou,Greece
[c] Department of Informatics, University of Piraeus, 18534 Piraeus, Greece
[d]Department of Engineering Systems and Services, Delft University of Technology (TU Delft), 2600 AA Delft, The Netherlands

Abstract

Enterprise Information Systems have a long-established and crucial role for modern organizations, as they enable seamless integration and management of critical business processes, ensuring efficiency in operations, data accuracy, and enhanced decision-making capabilities. One of their most interesting emerging technologies refer to the use of Artificial Intelligence as they may seamlessly automate routine tasks, offer predictive analytics, and provide deep insights, ultimately leading to intelligent data-driven decisions and improved operational efficiency. Of course, this direction of work is accompanied by some important challenges that come from the opacity of certain AI models and their potential biases due to low-quality training data used. In this paper, we argue that such challenges can be mitigated by a novel framework able to integrate, in a transparent manner, quality-related metadata on datasets used for training the AI-enabled emerging technologies in the field of EIS systems. These metadata are minted as Non-Fungible Tokens (NFTs) over the blockchain.

* Corresponding author. Tel.: +302108203586.
E-mail address: jverg@aueb.gr

10.1016/j.procs.2025.02.153

Keywords: NFT-based Data Provenance; AI Transparency in EIS

1. Introduction

The integration of Artificial Intelligence (AI) into Enterprise Information Systems (EIS) presents a significant opportunity for modern organizations to enhance their operational efficiency and innovate through emerging technologies. However, the adoption of AI in EIS introduces several challenges that mainly originate from concerns regarding the opacity of certain AI models and their potential biases [1]. Particularly, this concerns the trustworthiness and quality of the datasets used for machine learning. Ensuring that datasets are unbiased, reliable and of good quality is crucial for the accurate and ethical deployment of AI systems, especially those introduced as emerging technologies that aspire to significantly enhance the EIS systems of modern organizations.

One of the main success factors in enhancing EIS systems with AI is that the datasets for the training of AI models need to be reliable and any quality checks have to be recorded in a transparent way. This will increase the adoption of EIS users. The trustworthiness of a dataset is achieved by considering many factors among which: the reputation of the data creator/owner, the confidence in the data sources used, the data quality, their adequate volume when used for training and last the lack of any bias. Bias detection in datasets used can lead to unfair, erroneous or even unusable outcomes. The bias prevalence in AI datasets has been highlighted in recent studies [2], [3], a fact that raises concerns about the ethical implications of using AI across different application domains. In addition, the actual data volume used in a training process also poses challenges. While large datasets are often necessary for training robust AI models, they can contain noisy, incomplete, or inaccurate data points that may jeopardize the quality of the output AI model. On the other hand, the size of such datasets can constitute manual verification cumbersome or even impractical in certain cases. This might lead to an ever-increasing need for developing or improving automated data quality assessment tools. All these aspects can be perceived as metadata of a certain dataset. It is also indisputable that such metadata are clearly heterogenous, domain specific and their evaluation may involve automated software-based assessment, human experts as evaluators or any such combination.

Despite the availability of several automated tools [1-4] (as discussed in Section 2) that mitigate even partially the cumbersome task of evaluating data quality before using it for generating AI models in EIS, there is still a critical need that remains unanswered. This need drives the main research question of this paper which is the following: *Can there be a framework able to integrate, in a transparent manner, the results of any automatic dataset evaluation solutions along with human experts' evaluations, to boost the trustworthiness of AI-enabled emerging technologies in the field of EIS systems?* This paper delves into the main challenges and research questions surrounding the use of AI in EIS, emphasizing the importance of dataset trustworthiness and on how it can be transparently registered for mitigating any EIS users' concerns. This is primary a data provenance problem as it refers to a documented and non-repudiated trail that accounts for the origin of a piece of data [4] which in our case represents important metadata for datasets used to train AI-enhancements in EIS. Therefore, we discuss a framework that can combine the efficiency of automated tools with the contextual understanding and judgment of human experts, immutably registering their "evaluations" as metadata over the blockchain. We argue that this immutable registration of evaluation results as metadata over the blockchain, it will enhance the transparency and accountability of AI use in EIS, providing an auditable trail of data quality assessments.

The rest of the paper is structured as follows: in section 2 we discuss relevant work, while in section 3 we describe our methodological approach towards a Non-Fungible Token (NFT)-based dataset provenance mechanism that will permit the transparent integration and immutable persistence of verified metadata about datasets used for training AI-enabled EIS. In section 4, we provide an architectural overview along with all the relevant implementation details of a proof of concept that materializes our proposal. Last, we conclude this work in section 5, discussing the next steps of this research work.

2. Relevant Work

The growing concerns on the opacity and potential biases of certain models in the field of Artificial Intelligence have been accompanied by several recent research efforts that try to mitigate these concerns by improving fairness and explainability in AI decision making [1]. In general, fairness in AI implies the use of adequate datasets that do not discriminate, directly or indirectly, against any sensitive (sub)populations. Datasets may contain bias of various forms that are not always easy to pinpoint and often are quite domain specific. Examples include datasets created on historical, labelled data that have been based on discriminative past decisions; underrepresentation bias might be inserted in a dataset in cases where particular subpopulations are not adequately represented in the respective distribution, etc. A plethora of recent research efforts is available, yielding a plethora of respective bias detection methods that are able to measure bias through statistical formulas on the labelled training data or on the model's outcomes on the deployment data [5], [6], [7]. In general, existing bias mitigation strategies are classified as pre-training (i.e. enhance the dataset to mitigate its inherent biases), training (i.e., reduce the discrimination that the model could learn from the data), and post-training approaches (i.e., correct the discrimination learned by modifying the output). For example, in [1] a post-training technique is proposed to create a mitigated bias dataset by using a mitigated causal model that adjusts cause-and-effect relationships and probabilities within a Bayesian network. Several other surveys like the one in [2] have indicated the extend of such challenges in AI and they have examined different domains and subdomains, presenting what researchers have observed regarding to unfair outcomes in the state-of-the-art methods and ways they have tried to address them. There are still many future directions and solutions that can be taken to mitigate the problem of bias in AI systems, but the part that hasn't been addressed so far, concerns the pre-training phase in which the use of datasets of proven quality that has been indisputably registered by a network of human experts and automated software-based solutions.

With respect to immutably registering any kind of information that should be available to multiple entities, the Distributed Ledger Technology (DLT) or just blockchain, has been considered for several years the most appropriate solution. There are hundreds of decentralized digital ledgers that can record and verify transactions across a network of "miners" and therefore register blocks containing a cryptographic hash of the previous block, a timestamp, and recent transaction data. The consensus algorithm used by each DLT essentially provides a secure and transparent way to record transactions, ensure their immutability and consistence [8]. An important part of a subset of these DLTs is the deployment and execution of smart contracts (e.g., in Ethereum), which refer to executable code on the blockchain. The immutable nature of DLTs ensures that once a contract is deployed, its terms cannot be altered, thus fostering transparency and trust among all entities involved. A special type of smart contracts are the NFTs that recently gained momentum in the art world. NFTs refer to unique digital tokens registered in a blockchain that cannot be copied, replaced, or subdivided that are used to certify the authenticity and ownership of a digital or physical object. In this paper, we have decided to use these special types of smart contracts in order to distinctively characterize datasets in a transparent and indisputable manner, by being able to programmatically verify datasets' quality, origin, and suitability for machine learning training tasks.

3. Non-Fungible Token (NFT)-based dataset provenance mechanism

In this work, we adopt the concept of Non-Fungible Tokens (NFT) as a provenance mechanism for various datasets provided or generated within the context of an EIS system. The NFT-based Data Provenance Mechanism will be responsible for attaching a wide range of metadata (e.g., unbiased data, adequate volume, appropriate data quality etc.) on each dataset and minting them transparently and immutably, over a permissioned and private blockchain. These metadata are defined per dataset by either human evaluators or dedicated software services or both, to create a trustworthy description of the datasets that can be used for machine learning with respect to emerging technologies that enhance EIS systems. Additionally, several data provenance-related information (e.g., dataset owner(s), parent dataset(s) used as input etc.) can be attached, as well.

We consider NFTs as a way to attach descriptive metadata to datasets that will be used for machine learning purposes in a transparent, immutable and unique way defined in smart contracts. Our goal is to enhance specific datasets with metadata by multiple authorized entities in a decentralized manner for provenance, safety and auditability reasons. These metadata allow datasets to be identified, discovered and associated with specific characteristics that will be evaluated later on by data analysts who would like to explore or use it in their enhanced EIS systems.

The choice of implementing an NFT-based dataset provenance approach, compared to a classic web2.0 metadata management approach, has the advantage of an immutable, transparent and safe persistence of metadata. The process involved is defined in a smart contract and the protocol implemented by the nodes participating in the blockchain network instead of being dictated by a single server.

We have designed and implemented a decentralized prototype application based on a smart contract that defines a process for creating ("minting" is the common term for creating a new NFT token) a new unique token representing a dataset when specific requirements are met. This involves a specific number of attributes, defined by data engineers, which are evaluated by multiple authorized entities participating in the blockchain network. When the evaluation process is complete, the dataset owner can mint the uniquely identified NFT. All the information associated with this unique object is stored on chain and the whole minting process is transparent to everyone participating in the network. Whenever a transaction, that changes the "state" of the blockchain network, takes place, appropriate messages (events in the "smart-contract" world) are generated and broadcasted among the nodes of the network, so that the corresponding actions regarding the application interaction can be made. In our case, this involves informing the users through a user interface (UI) about the actions made. The sequence diagram depicted in Figure 1 illustrates the implemented approach for minting an NFT.

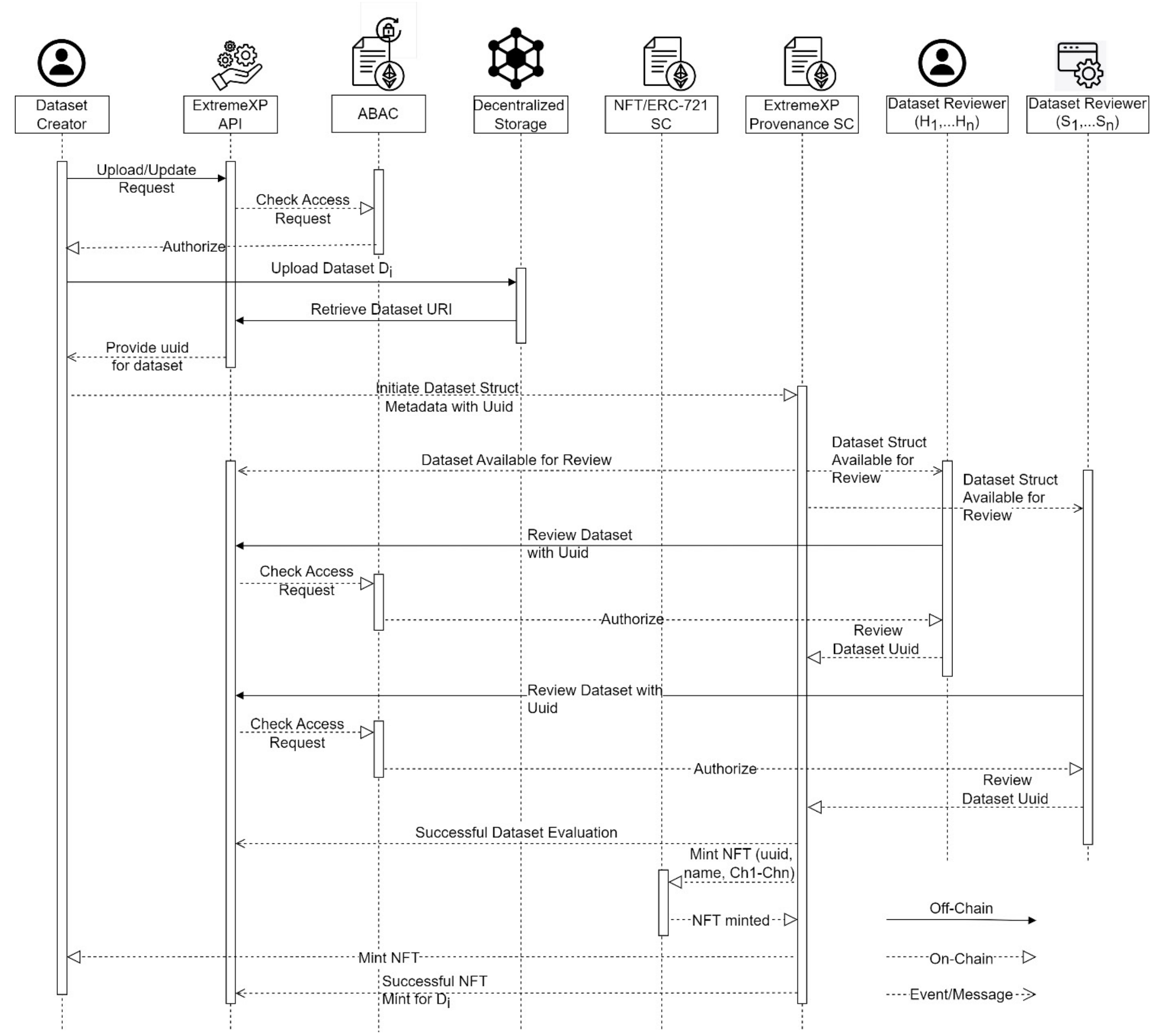

Fig. 1. Overview of the NFT-based provenance mechanism - Minting an NFT

The Dataset Creator requests access to mint an NFT. At this stage an authorization step is needed which in our work follows the advanced Attribute-based Access Control (ABAC) paradigm [9]. This mechanism is invoked through an API call and returns the authorization result. Further details on this mechanism are out of scope of this paper. If authorization is granted, the Dataset Creator uploads the dataset itself and retrieves the corresponding URI (Uniform Resource Identifier). As a next step, she interacts with a dedicated smart contract, called ExtremeXP Provenance SC and calls the add function by providing the URI as argument. This function call initializes a new structure in the smart contract for the uploaded dataset, within which any metadata is stored. As soon as the transaction gets confirmed on the blockchain network, an event message is broadcasted and returned to the dataset creator, as well as to the Dataset Reviewers (Humans: H1...HN and software Services: S1...SN). Afterwards, dataset reviewers interact with a UI and request access to evaluate a dataset. The ABAC mechanism is invoked again through an API call and returns the authorization result. If authorization is granted once more, reviewers provide their evaluation values for the attributes they are authorized to evaluate. Every time a new evaluation is added, an event message is emitted, and all parties get informed about it. When the evaluation is complete, the data owner can mint the NFT for the uploaded dataset. A final event message is emitted, and all parties get informed about the creation of the new NFT.

As far as the retrieval of a dataset is concerned, the process is demonstrated in the next sequence diagram (Figure 2). At first, a domain expert or a data analyst interacts with the UI and requests a Dataset or a list of Datasets. As always, the ABAC mechanism is invoked through an API call and returns the authorization result. If authorization is granted, the associated metadata and provenance information are resolved and returned. As a final step, the user requests to get access to the desired dataset and retrieves it from the decentralized storage component.

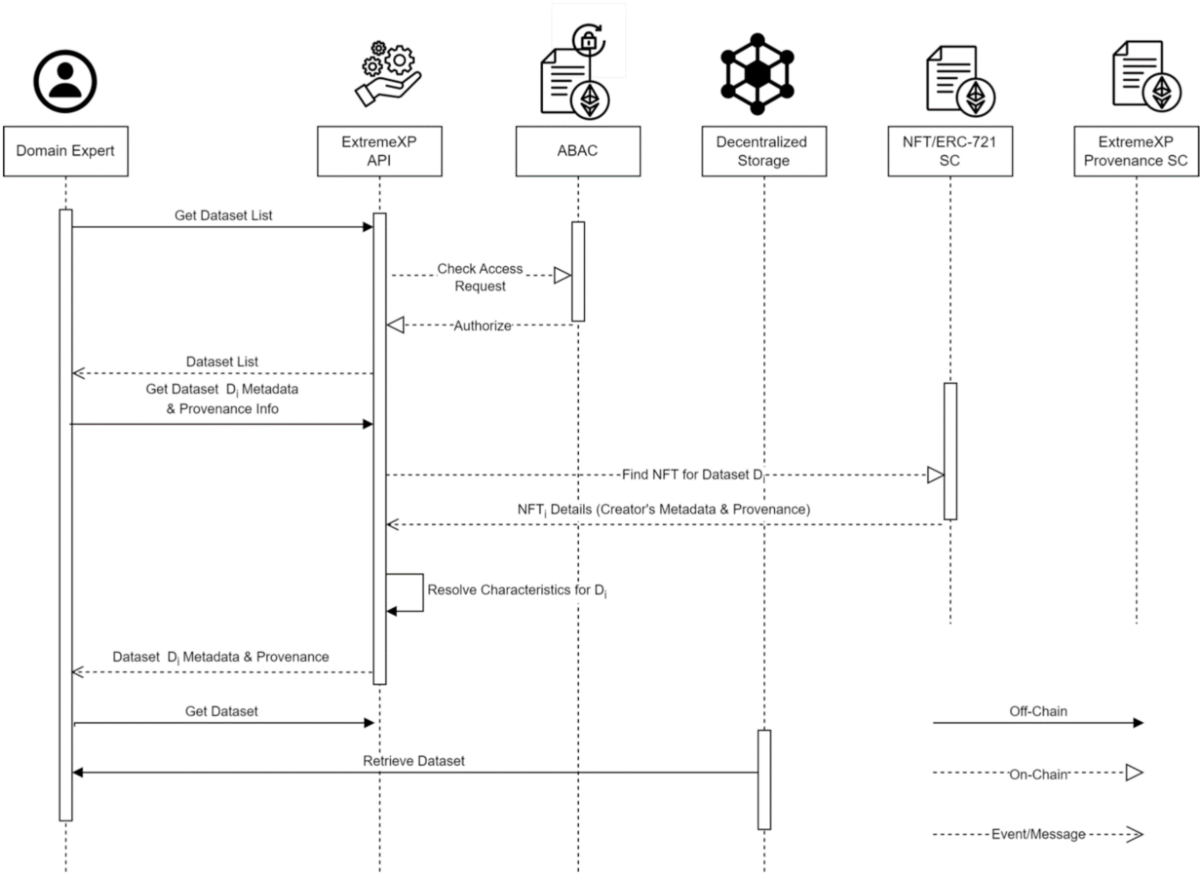

Fig. 2. Overview of the NFT-based provenance mechanism – Retrieving a Dataset

4. NFT-based Data Provenance Proof of Concept

The detailed architectural overview of the NFT-based Data Provenance application is provided in the following figure along with implementation specifics. Libraries used in each component and interactions between components are also noted. Both approaches are implemented using ERC721 as a base NFT contract, extended with enumerable or URI storage functionalities provided by OpenZeppelin libraries [10]. The proof of concept is available in the following Gitlab repository.†

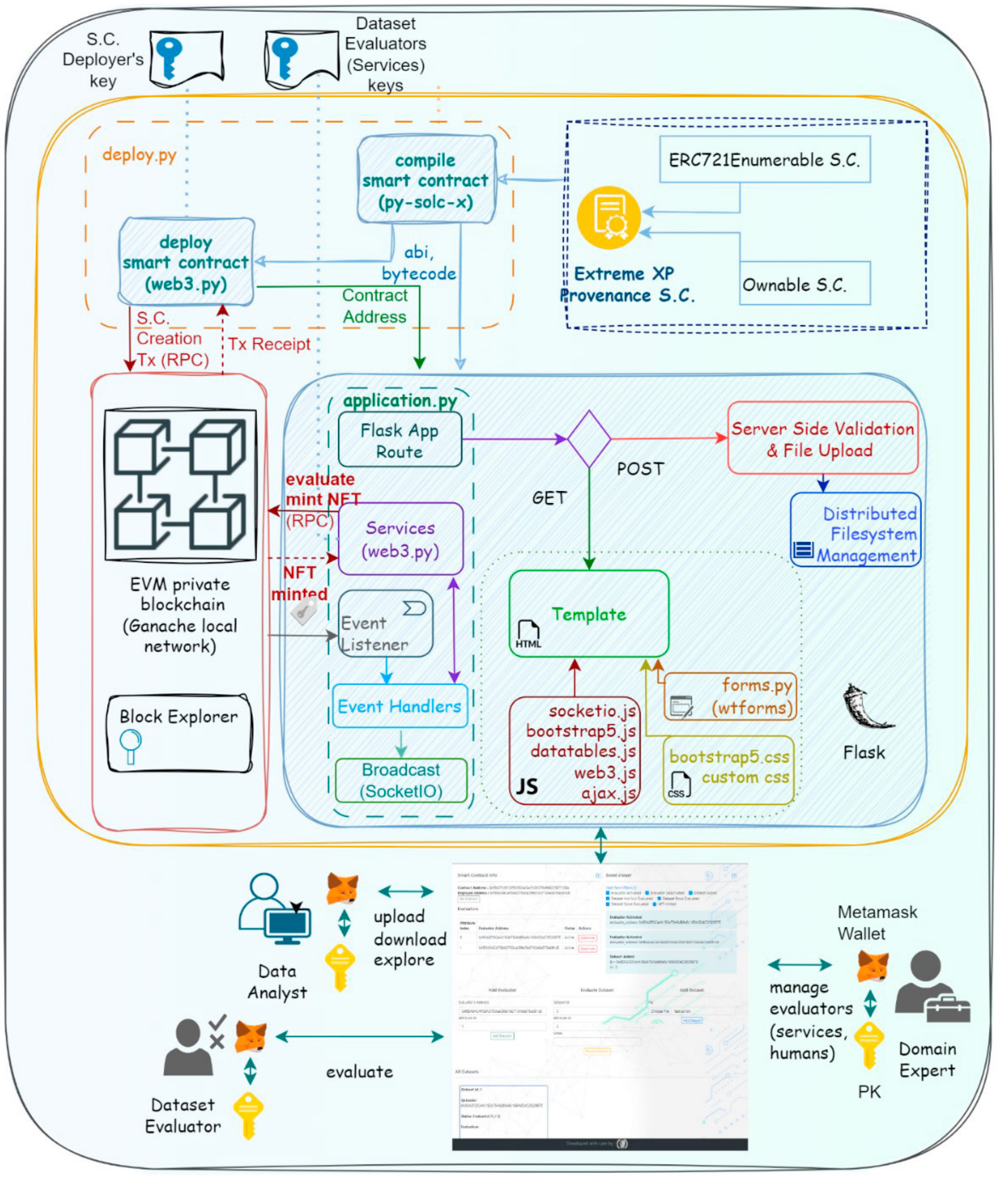

Fig. 3. Architectural overview of the NFT-based Data Provenance application

† https://github.com/IMU-ICCS/nft_based_provenance/tree/main

We developed the smart contracts, depicted in Fig.3, in Remix IDE [11], which is an open-source web and desktop application that provides an integrated development environment (IDE) for smart contract development on Ethereum. It has a built-in Solidity compiler and a testing environment, which enables developers to compile their smart contracts without the need for external tools. This helps to ensure the correctness and reliability of the code. Additionally, we set up a Ganache blockchain network [12] to demonstrate its use. Ganache is a personal blockchain for Ethereum development that is used to deploy contracts, develop decentralized applications (DApps), and run tests. Ethereum was chosen as the primary blockchain for this project not only because it is the most widely used and trusted platform, but also because of its strong foundation in NFTs.

Specifically, we developed a python script (deploy.py), which undertakes the following important tasks, as it:

- compiles the solidity code using a python wrapper and version management tool for the solc Solidity compiler by configuring the appropriate solidity version of the smart contracts (py solcx [13]).
- returns the application binary interface (ABI) to application. In general, an ABI is the interface between two program modules, one of which is often at the level of machine code. The interface is the de facto method for encoding/decoding data into/out of the machine code. It consists of a list of the contract's functions and arguments (in JSON1 format). An account wishing to use a smart contract's function uses the ABI to hash the function definition, so it can create the EVM bytecode required to call the function.
- deploys the compiled smart contract on Ganache network via a Remote Procedure Call (RPC) using web3.py [14], a Python library for interacting with Ethereum. It provides a Python interface to the Ethereum blockchain that allows developers to interact with Ethereum nodes, send transactions, deploy and interact with smart contracts. Furthermore, it allows to query information about the blockchain, such as block details, transaction information, and account balances as well as to call methods on smart contracts, send transactions, and retrieve contract data. Moreover, it supports event handling, allowing our application to listen and react to events emitted by Ethereum smart contracts. Finally, it also provides tools for managing transaction gas prices, nonce, etc.
- receives the transaction receipt, extracts smart contract's address and returns it to the application.

Having deployed the smart contract on the Ganache network, we developed an application with a flask web server that delivers a UI for users to interact with the deployed smart contract [15]. Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries and is designed to be simple yet extensible with several key features (e.g., routing, RESTful API Support, Werkzeug and Jinja2 etc.). Making use such key features, our application supports the following main functionalities. It:

- serves template with associated JavaScript libraries which allow interactions with the smart contract. Key libraries are web3.js (same functionality as web3.py described above), SocketIO (allows socket connections for real-time event-broadcasting) and Ajax (allows cross-site requests using FETCH API)
- Delivers API endpoints as a middleware for managing Distributed File System (if required)
- Provides internal services for automated evaluation of datasets in Provenance smart contract
- Provides an Event Listener which allows the subscription to smart contract's events. This functionality is implemented using a daemon process.
- Handles Events received by the event listener and triggers the execution of services or broadcasts corresponding messages to Application's clients.

Currently, both prototype implementations use a simple authorization mechanism, which allows data engineers to give or revoke permission to certain entities to evaluate datasets about specific attributes. In addition, in our prototype we use the InterPlanetary File System (IPFS) for off-chain storage, besides the on-chain one, to persist all the NFT related metadata, as well as for the datasets themselves [16]. Every object added to an IPFS network gets assigned a unique cryptographic hash (a digital fingerprint of sorts) that represents its contents exactly. These identifiers are based on the content itself rather than its location. Consequently, duplicate files result in identical hashes, it is thus assured that each hash points to one distinct piece of information. Last, our smart contract is designed with a robust indexed event logging system to ensure transparency and facilitate efficient data tracking.

5. Conclusions

This paper introduced a novel framework that can integrate, in a transparent manner, the results of any automatic dataset evaluation solution along with human experts' evaluations, in the form of metadata minted as NFT tokens over the blockchain. This transparent and non-repudiable recording of evaluation metadata on AI training datasets aims to boost the trustworthiness of AI-enabled emerging technologies in the field of EIS systems. We presented the methodological framework along with the architecture and technical implementation details of a first prototype. Specifically, UML sequence diagrams were provided to illustrate the introduced approach for minting and retrieving NFTs to transparently and immutably attach metadata on training datasets. In addition, the architectural view and implementation details of the proposed NFT-based Data Provenance application were provided. The prototype follows the ERC721 standard, and it was deployed over the Ganache private Ethereum blockchain network.

Next steps of this work involve first an evaluation of the first prototypical implementation and the development of a second prototype, based on biddings to facilitate gas-efficiency. Furthermore, we will examine the research and extension of the NFT-based dataset provenance mechanism with a reward/reputation mechanism for parties involved in the Provenance smart contract's transactions (data creators and evaluators). As far as central entities are involved in the current implementation and with respect to the chosen blockchain network, we also intend to investigate whether they can be replaced by a committee of oracle nodes, who will be responsible for bringing on chain information, registered and exposed in the web2.0 world [17]. Last, we intend to explore frameworks like Chainlink's CCIP or LayerZero to vest on the potential benefits of broader cross-chain interoperability by enabling communication across different blockchain ecosystems. This will enhance flexibility, allowing datasets to be minted, evaluated, and traded across multiple chains, while maintaining the integrity of the NFT tokens.

Acknowledgements

The research leading to these results has received funding from the EU's Horizon Europe research and innovation programme under grant agreement No. 101093164 ***ExtremeXP*** *project.*

References

[1] Rubén González-Sendino, Emilio Serrano, Javier Bajo. (2024) "Mitigating bias in artificial intelligence: Fair data generation via causal models for transparent and explainable decision-making." Future Generation Computer Systems 155, pp. 384-401, https://doi.org/10.1016/j.future.2024.02.023

[2] Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2022) "A Survey on Bias and Fairness in Machine Learning." ACM Computing Surveys, 54(6), 1-35, https://doi.org/10.1145/3457607

[3] Barocas, S., Hardt, M., & Narayanan, A. (2023) "Fairness and Machine Learning." ISBN: 9780262376525, MIT Press.

[4] Karl Werder, Balasubramaniam Ramesh, and Rongen (Sophia) Zhang. (2022) "Establishing Data Provenance for Responsible Artificial Intelligence Systems." ACM Trans. Manage. Inf. Syst. 13, 2, Article 22 (June 2022), 23 pages. https://doi.org/10.1145/3503488

[5] Sahil Verma and Julia Rubin. (2018) "Fairness definitions explained." In Proceedings of the International Workshop on Software Fairness (FairWare '18). Association for Computing Machinery, New York, NY, USA, 1–7.

[6] Loukas Kavouras, Konstantinos Tsopelas, Giorgos Giannopoulos, Dimitris Sacharidis, Eleni Psaroudaki, Nikolaos Theologitis, Dimitrios Rontogiannis, Dimitris Fotakis, Ioannis Emiris. (2023) "Fairness Aware Counterfactuals for Subgroups." NeurIPS 2023

[7] Dimitris Sacharidis, Giorgos Giannopoulos, George Papastefanatos Kostas Stefanidis. (2023) "Auditing for Spatial Fairness." EDBT 2023

[8] [Wood G. (2014) "Ethereum: A secure decentralised generalised transaction ledger." Ethereum Project Yellow Paper, 151, 1-32.

[9] Tuler De Oliveira, M., Reis, L. H. A., Verginadis, Y., Mattos, D. M. F., & Olabarriaga, S. D. (2022) "SmartAccess: Attribute-Based Access Control System for Medical Records Based on Smart Contracts." IEEE Access, 10, 117836-117854. https://doi.org/10.1109/ACCESS.2022.3217201.

[10]Openzeppelin- ERC 721 contracts, available online at: https://github.com/OpenZeppelin/openzeppelin-contracts/tree/master/contracts/token/ERC721

[11] Remix IDE, available online at: https://remix.ethereum.org/

[12] Ganache, Truffle Suite, available online at: https://trufflesuite.com/ganache/

[13] Python wrapper and version management tool for the solc Solidity compiler, available online at: https://pypi.org/project/py-solc-x/

[14] Web3.py: A Python library for interacting with Ethereum, available online at: https://pypi.org/project/web3/

[15] Flask framework, available online at: https://flask.palletsprojects.com/en/3.0.x/

[16] Protocol Labs. (n.d.). IPFS - InterPlanetary File System. Retrieved [10/07/2024], from https://ipfs.io/

[17] Oracles in Solidity, available online at: https://ethereum.org/en/developers/docs/oracles/