The Model Openness Framework: Promoting Completeness and Openness for Reproducibility, Transparency, and Usability in Artificial Intelligence

Anonymous Author(s)
Affiliation
Address
email

Abstract

Generative artificial intelligence (GenAI) offers numerous opportunities for research and innovation, but concerns have been raised about the reproducibility, transparency, and safety of frontier AI models. Many "open-source" GenAI models lack the necessary components for full understanding, auditing, and reproducibility, while some models use restrictive licenses, a practice known as "openwashing". In this paper, we propose the Model Openness Framework (MOF), a three-tier ranked classification system that rates machine learning models based on their completeness and openness. Each MOF class specifies the code, data, and documentation components in the model development lifecycle that should be released under certain open licenses. We develop the Model Openness Tool (MOT) to provide a user-friendly reference implementation to evaluate models' openness and completeness against the MOF. We launched the Open MDW License recently, which is the first permissive open license for AI models. The MOF aims to establish completeness and openness as core tenets of responsible AI research and development, and to promote best practices in the burgeoning open AI ecosystem.

1 Introduction

Generative artificial intelligence (GenAI) has seen remarkable advances in recent years [31], however, concerns have also grown regarding its transparency, reproducibility, and safety [6, 67, 4]. Many state-of-the-art models are closed and accessible only through APIs, making it difficult to explain the inner workings and ensure fairness. As an alternative, many companies, researchers, and individuals release AI models publicly on platforms such as Hugging Face, GitHub, and Kaggle [52, 7, 50]. It indicates a growing momentum towards open AI models.

However, there are major concerns regarding models' completeness and openness. First, many model producers do not release key artifacts throughout the development lifecycle. They only release selected artifacts, e.g., model architecture & parameters. Without the full availability of datasets, training code, and detailed documentation, it is difficult to reproduce/validate/audit the models. Second, licensing practices further undermine openness. Many models are released under restrictive licenses or inappropriate open-source licenses (designed for conventional software). They are falsely promoted as "open-source", a practice called "openwashing" [44, 68, 33, 32]. This can mislead downstream users, limit usability, and expose them to legal risks. This lack of transparency and reproducibility hinders real-world deployment in industry and potentially erodes trust in AI [24, 53].

Some recent initiatives aim to facilitate the openness of AI models, such as the Open Source Initiative's first version of Open Source AI Definition [47] and the Mozilla Foundation's openness framework across the AI stack [3] (which was partially inspired by our work). However, they do not evaluate both the completeness and openness of models. The EU AI Act [15] focuses more on the legal compliance of AI instead of a practical guideline for AI model distribution.

Submitted to Workshop on Regulatable ML at the 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

MOF Class	Components Included	Usage
Class I. Open Science	Research Paper Datasets Data Preprocessing Code Model Parameters (Intermediate Checkpoints) Model Metadata (Optional) All Class II and III Components	End to end analysis and auditing Reproduction of a similar model Data exploration and experimentation
Class II. Open Tooling	Training, Validation, and Testing Code Inference Code Evaluation Code Evaluation Data Supporting Libraries & Tools All Class III Components	Understand training process Validate benchmark claims Inference optimizations
Class III. Open Model	Model Architecture Model Parameters (Final Checkpoints) Technical Report or Research Paper Evaluation Results Model Card Data Card Sample Model Outputs (Optional)	Unrestricted usage (access, use, modify, redistribute) Create a product or service Fine tune and align Model optimizations

Figure 1: Classes and components of the MOF. Class III represents the minimum level of completeness, while Class I represents the highest. Each class builds upon the previous ones.

In this paper, we propose the Model Openness Framework (MOF) for evaluating and classifying the completeness and openness of machine learning models across their development lifecycle. We also develop the Model Openness Tool (MOT) to provide a practical, user-friendly way for model producers to apply the MOF. It currently hosts the evaluation of 235 models, providing the details of their MOF classes and licenses. An important milestone is the recent launch of the Open Model, Data, and Weights License Agreement (OpenMDW V1.0), which is the first open license for machine learning models and their related artifacts. The MOF aims to establish completeness and openness as core tenets of AI R&D, promoting transparency, reproducibility, and usability in AI. Its adoption can foster a more open, transparent, and responsible AI ecosystem.

The remainder of this paper is organized as follows. It begins with the three classes of the MOF classification system. Then, it defines the 16 model components and the MOF configuration file, each with acceptable licenses. Next, it discusses the practical adoption, benefits, and limitations of the MOF. It concludes with a summary of the key contributions.

2 Overveiw of Model Openness Framework

2.1 MOF Structure

The MOF proposes a three-tier classification system to classify the degree of completeness and openness of ML models across all aspects of a model's development lifecycle, as shown in Fig. 1. The MOF has 17 components to fulfill the completeness of model artifacts, i.e., 16 components and 1 MOF config file. The 16 components cover the code, data, and documentation along the model development lifecycle. The distribution includes an additional component, the MOF configuration file, to comply with the MOF requirements.

The 16 model components are categorized into three distinct classes, where model parameters are further split into final checkpoints and intermediate checkpoints. Each class builds upon the previous one, with Class III being the least complete and Class I being the most complete. There is an inclusion relationship between classes, where Class II includes all components from Class III, and Class I includes all components from both Class II and III. The higher the class indicates the more complete and open distribution that promotes more transparency and enables reproducibility, auditing, and downstream use. This approach is more meaningful than a calculated index, as it guides model producers in providing essential components released under open licenses for each tier of the framework. As the class of the MOF increases, the producer moves closer to a more complete distribution that best aligns with the principles of open science in AI. To qualify for a particular class, the producer must provide every required component for that class, released under an appropriate open license from Fig. 2.

2.2 Three Classes of MOF

The three classes of the MOF represent ascending levels of model completeness and openness.

Class III. Open Model. Class III is the entry point and contains the minimum required components that must be released using open licenses. If not all of these components are included in a release and not all components use an open license, then the entire release cannot be considered open under the MOF. The Open Model class covers: 1). Core model architecture and the final set of parameters; and 2). Light documentation conveying capabilities and characterization of the model and data.

Class III contains components required to study, modify, redistribute, and build upon a model without restrictions, including commercial and educational purposes. The inclusion of the model architecture, final weights and biases, and documentation (e.g., the technical report, evaluation results, model, and data cards) provides the necessary information to work with the model and understand its capabilities, constraints, and the nature of the training data. However, this class lacks completeness and robustness for full reproducibility and the transparency needed to confirm all claims made by the producer.

Class II. Open Tooling. Building upon Class III, Class II provides model consumers with the complete codebase including libraries/tools needed for training and testing models. Added elements include: 1). Full training-inference code; 2). Benchmark tests to validate and quantify performance; and 3). Libraries/tools to ease integration and to complete the codebase (optional).

This tier is an intermediate step between an open model and open science, providing a model consumer with information to test a model producer's assertions. It also allows a model consumer to perform debugging and model enhancements. Although it does provide insights into the training process, it does not include the actual datasets. It is also lighter on documentation, which limits a deeper understanding of the model's intricacies.

Class I. Open Science. The top tier aligns with the ideals of open science: the sharing of all artifacts needed for end-to-end transparency, reproducibility, and collaboration. This includes: 1). A detailed research paper conveying the genesis of the model and its evolution; 2). Raw training datasets used in the training of the model (any license or unlicensed); 3). Checkpoint weights showcasing full model evolution; and 4). Log files providing yet more low-level insights. Fulfilling Class I empowers the community to inspect models through the model lifecycle, representing the gold standard for completeness and openness rooted in scientific principles.

3 MOF Components and Acceptable Licenses

This section specifies the 16 model components and the MOF configuration file. They cover the degree of completeness and openness across all aspects of the development process, including training data, model architecture, model parameters, evaluation benchmarks, and documentation. The content type of each component is classified as data, code, or documentation, as shown in Fig. 2. The table specifies standard open licenses that should be used for releasing each component while allowing some flexibility for equivalent licenses.

Note that not all components are required for all classes. Each component section below specifies the classes that it applies to, consistent with Fig. 1. Note that not all components need to be distributed separately; some MAY be combined. E.g., evaluation results MAY be included in a research paper, technical report, or model card rather than published as a standalone artifact.

3.1 Model Architecture (III.1)

The model architecture is the core of any ML project. It can include the ML algorithms, neural network layout, connectivity, activations, and other architectural elements. While the model architecture is often closely tied to the trained model parameters, sharing the architecture alone allows others to understand the structure of the model without necessitating the release of the fully trained model. The model architecture should be fully described in the paper and shared as open-source code. This enables implementation, analysis, extensions, adaptations and unrestricted usage of the model or models. The model architecture is a code artifact and to be considered open, must be released under an OSI-approved open-source license that does not limit its usage and derivative works.

MOF Class	Component	Content Type	Accepted Open License	
			Preferred	Acceptable
III.1	Model Architecture	Code	/	OSI-approved
III.2	Model Parameters (Final)	Data	CDLA-Permissive-2.0	Permissive Open Data Licenses
III.3	Technical Report	Documentation	CC-BY-4.0	Permissive Open Content Licenses
III.4	Evaluation Results	Documentation	CC-BY-4.0	Permissive Open Content Licenses
III.5	Model Card	Documentation	CC-BY-4.0	Permissive Open Content Licenses
III.6	Data Card	Documentation	CC-BY-4.0	Permissive Open Content Licenses
III.7	Sample Model Outputs	Data or Code	/	Unlicensed
II.2	Training Code	Code	/	OSI-approved
II.3	Inference Code	Code	/	OSI-approved
II.4	Evaluation Code	Code	/	OSI-approved
II.5	Evaluation Data	Data	CDLA-Permissive-2.0	Permissive Open Data Licenses
II.6	Supporting libraries and Tools	Code	/	OSI-approved
I.2	Research Paper	Documentation	CC-BY-4.0	Permissive Open Content Licenses
I.3	Datasets	Data	CDLA-Permissive-2.0	Any including unlicensed
I.4	Data Preprocessing Code	Code	/	OSI-approved
I.5	Model Parameters (Intmd.)	Data	CDLA-Permissive-2.0	Permissive Open Data Licenses
I.6	Model Metadata	Data	CDLA-Permissive-2.0	Permissive Open Data Licenses

Figure 2: Components and licenses of the Model Openness Framework. Each component is one of three content types (data, code, and documentation) and requires appropriate open licenses. We show 17 components because the model parameters are split into final checkpoints and intermediate points.

3.2 Model Parameters – Final Checkpoints (III.2)

Trained model parameters must be released under an open license. In the case of deep learning models, checkpoints from key intermediate training stages, as well as the final optimizer state, should be included. At a minimum, the final model parameters and optimizer state (when applicable) must be distributed, whether compressed or uncompressed, in a format compatible with popular deep learning frameworks such as TensorFlow, Keras, PyTorch, or the framework-independent ONNX file format.

To date, model producers have been releasing model parameters (i.e., weights and biases) using an open source license, such as Apache 2.0 and MIT, even though model parameters are not compatible with such licenses. Since model parameters are in fact data, model parameters should be distributed under an open data license, like CDLA-Permissive-2.0. Although licenses designed for open source software are permissive and indemnify the developer from liability, open data licenses are better suited to data-specific considerations such as privacy, ethics, and data rights. Most permissive licenses do not refer to data directly and do not address the ability to modify and redistribute model parameters. This gap could result in a legal obligation to any model consumer if the model producer were to implement royalties after the widespread adoption of their model. This is a legal gray area that remains untested. The model architecture and model parameters should be distributed separately, as each one requires a different type-appropriate open license. This separation allows each component to be studied, modified, redistributed, and used independently of the other.

3.3 Technical Report (III.3)

The technical report is less detailed than a research paper. It provides necessary documentation for the model consumer to understand performance, usage, and implications, but not enough to reproduce the model. The technical report is optional if a research paper is included. The goal is to characterize model capabilities and provide adoption and impact guidance. The technical report must be released under an open license for documentation, ideally CC-BY-4.0 or CC0, on an open access platform, and must be included in the distribution for permanence.

3.4 Evaluation Results (III.4)

Evaluation results, including quantitative metrics and results from model evaluation, must be reported in the research paper or technical report. Tests can evaluate factors such as model efficiency, accuracy, performance, fairness, bias, toxicity, and truthfulness. Producers must include benchmark test results, whether industry standard or custom-developed. For industry standard benchmarks, the test suite name, test name, and version number must be included with the results. Custom benchmarks, whether in code or any form of media, must be included in full for validation. The evaluation results should be summarized in the technical report and research paper, depending on the MOF class. Raw outputs of the model evaluation should be distributed for easy verification, using an open license like CC-BY-4.0.

3.5 Model Card (III.5)

A model card provides metrics, usage guidance, and details about a model [39]. Model cards should cover model details, intended uses, factors, evaluation, risks, and mitigations related to the model. This provides transparency into model behavior. The model card itself must use a permissive license that covers documentation, ideally CC-BY-4.0.

3.6 Data Card (III.6)

A data card provides summary statistics and key information about a dataset to enhance understanding of its composition [17]. Following guidelines from the Data Nutrition Project, data cards should describe various aspects of the dataset, including the features, instances, intended uses, motivation, and collection process. Data cards help identify potential biases in datasets and guide proper usage by downstream users. They also contribute to reproducibility and transparency by detailing the entire data preparation process. The data card must be released under a permissive license that covers documentation, with CC-BY-4.0 being an ideal choice.

3.7 Sample Model Outputs (III.7)

Sample model outputs are an optional component. If they are included in the distribution, they must be shared publicly without copyright or restrictions, where legally permitted, to allow for redistribution with the release. These outputs can take various forms, such as text samples, images, videos, software code, audio, 3D assets, metadata, or any other potential output generated from the model, including predictions and probabilities. In certain sensitive domains, generated examples can be anonymized or simulated if needed. Sample model outputs help others perform a quick evaluation of the model's performance and provide a glimpse into its capabilities. If the model outputs are not copyrightable, they should be released without a license, and this should be noted in the LICENSE file. It is important to note that while sample model outputs are recommended, they are not a requirement for the MOF. Additionally, the MOF does not consider the actual model outputs generated by the model consumer during inference.

3.8 Training, Validation, and Testing Code (II.2)

The full code for training, validating, and testing the model should be open-sourced, including model construction, training loop, hyperparameter selection, and checkpointing. Any fine-tuning code, reinforcement learning code, or methods that modify model parameters or implement adapters affecting model performance must be included. This enables reproducible end-to-end training. Comments explaining the approach should be included, ideally following PEP 8 style guide for Python code. Including log files generated during training provides deeper insights and is recommended. The training, validation, and testing code must be released under an OSI-approved open-source license, while log files should use a permissive open-content license like CC-BY-4.0.

3.9 Inference Code (II.3)

Code for performing inference with the trained model must be shared under an open-source license. This includes any data preprocessing or postprocessing required during inference. It can include any model optimizations and dependencies like external libraries. It fundamentally includes any code required to fully replicate the benchmark results presented in the research paper for the project. The availability of inference code facilitates complete replication of the performance of the model, and it informs the model consumer about how to use the model most effectively for their applications. The inference code must be released under an OSI-approved open-source license.

3.10 Evaluation Code (II.4)

Evaluation code, evaluation data, and evaluation results are separate components in the MOF. This is due to the fact that some benchmarks are written in code and others only use data, for instance text used to evaluate an LLM or images used to evaluate a computer vision model. Many benchmark tests are a combination of both code and data used to evaluate a model, which includes the scripts needed to load the data and run benchmark tests. Since code and data require different licenses, they are separate components. Depending on the nature of the model and the methods used to evaluate it, the distribution may include one or both of evaluation code and data. Any code used for model evaluation and benchmarking must be included and distributed under an OSI-approved open-source license.

3.11 Evaluation Data (II.5)

When a model is evaluated using data, such as text, images, videos, audio, or 3D data, the evaluation data must be included in the distribution. However, if the model is not evaluated with data, then including the evaluation data is not necessary. In cases where the model producer relies on widely disseminated standard benchmark tests, it is sufficient to describe them in the technical report and whitepaper, along with the version of the test, rather than including them in the distribution. If the evaluation data is included in the distribution, it must use a permissive license appropriate for data or content, such as CDLA-Permissive-2.0, CC-BY-4.0, or CC0.

3.12 Supporting Libraries and Tools (II.6)

Supporting libraries and tools are an optional component. Releasing supporting code libraries, utilities, or tools developed in the course of the research under an open-source license makes them available for wider use. This could include data loaders, visualization code, simulation environments, etc. The use of existing and custom open-source tools should also be documented. Other tools and libraries may include:

- Software libraries and frameworks used in model development, along with version details.
- Tokenizers: Code used to tokenize text and any data used to train the tokenizer (if used).
- Hyperparameter search code: Code for automating hyperparameter tuning (if used).
- Compute infrastructure code: If specialized compute infrastructure was built to scale training, the setup code could be released.
- Monitoring code: Code for tracking experiments, metrics, artifacts, etc., during model development is often useful to open source as well.
- Containerization files: Dockerfiles or other container packaging to distribute the model could be shared.
- Frontend/visualization: Any web/mobile frontends or visualizations built on top of the model outputs could be released as open source.
- Deployment orchestration: Infrastructure-as-Code templates for deploying the model to production.
- Model integration code: Wrapper code/SDKs to integrate the model into downstream applications.
- Interactive demos: Links to hosted interactive demos of the model through Jupyter, Streamlit, etc.

Most libraries and tools will already have a license, so only if the model producer creates their own libraries or tools would they need to include them with the distribution and use an OSI-approved license for the software.

3.13 Research Paper (I.2)

The research paper details the model methodology, results, and analysis, following open science principles for accessibility and transparency. We suggest structuring the paper with an abstract, introduction, related work, methods, results, discussion, conclusion, and references. The paper must be released under an open license, ideally CC-BY-4.0, shared on an open-access platform like arXiv, and included in the model distribution.

3.14 Datasets (I.3)

Data is the lifeblood of ML models and is the most often held back element in the release of a model. Training data is data used for any form of model training including pre-training, fine-tuning, alignment using reinforcement learning techniques, or data used for other methods that otherwise modify the weights of the model. Datasets also include data used for model validation and testing, as well as data that may be used with benchmark tests. The datasets component may also include tokenized datasets when present. Data can be any form or combination of media, whether text, code, images, videos, audio, 3D objects, URIs, or any other data used for training, validation, and testing purposes. Datasets also include any metadata, from annotation data, such as labels, bounding boxes, and key points, to attribution, bitrates, resolution, and other metadata that may be relevant to a dataset used in the model development process.

The datasets used to develop the model ideally should be released under an open license allowing unrestricted access, modification, and reuse for any purpose, preferably Creative Commons CC-BY-4.0 or CC-0. We acknowledge that most pre-training data is subject to copyright, and therefore, it is not possible to license the data. To this end, datasets are an optional component, with the caveat that datasets must be included for Class I (with any or no license). Having access to the training data, whether pre-training, fine-tuning, alignment, or any other data, enables reproducibility and validation of the training process. Any limits on sharing due to privacy or sensitivity should be documented. It is preferable that both pre- and post-processed data are supplied. However, if this is not possible due to the size of the dataset, providing links to any curated raw datasets online is sufficient when accompanied by data preprocessing code.

3.15 Data Preprocessing Code (I.4)

The data preprocessing code is all the code used for preprocessing, cleaning, and formatting the training, validation, and testing data for a model. It also includes code used to transform fine-tuning data and code that is used for alignment tasks like Reinforcement Learning from Human Feedback (RLHF). Other data preprocessing code, such as code for data ingestion when appropriate, feature engineering, data augmentation, and tokenization, is also included. The data preprocessing code MUST be released using an OSI-approved open source software license.

3.16 Model Parameters – Intermediate Checkpoints (I.5)

In addition to the final checkpoints and optimizer states, for Class I models, the checkpoints and optimizer states (when applicable) from key intermediate stages of training, along with the log files, must be included and distributed under an open license. Intermediate model parameters SHOULD be distributed under an open data license, such as CDLA-Permissive-2.0.

3.17 Model Metadata (I.6)

Model metadata are an optional component. Model metadata refers to additional information about the model, beyond the model parameters and architecture, such as the version of the framework used to create it and custom tags or descriptions provided by the developer, including model and data lineage information. There is no particular requirement or profile for this type of metadata, and it can include any information the developer would like to provide with the shipped model. This metadata can be helpful for model management, especially when working with multiple versions of models or conducting experiments. Often the metadata is exported from or loaded by a metadata store. Any model metadata should use an open-data license such as CDLA-Permissive-2.0 to ensure it can be freely used and shared.

3.18 Model Openness Configuration File

The MOF configuration file is a crucial component of any model distribution, serving two primary purposes. It informs model consumers about the components included in the release; and it specifies the licenses under which each component is distributed. The MOF configuration file enables platforms that host models to understand the contents and licensing of the model distribution. The file itself is distributed under the Creative Commons CC-BY-4.0 license.

4 Adopting MOF in GenAI Models

This section discusses efforts to adopt the MOF. To assess its feasibility and help model producers apply the MOF, the Model Openness Tool (MOT) is developed. We also conducted a case study on DeepSeek models to evaluate their openness. To address the licensing challenges, the OpenMDW license is developed to cover ML models and associated artifacts under a single, permissive license.

Models	Overall Openness	Class III. Open Model							Class II. Open Tooling					Class I. Open Science				
		Sample Model Output (Optional)	Evaluation Results	Technical Report	Model Architecture	Model Parameters (Final)	Model Card	Data Card	Inference Code	Supporting Libraries & Tools	Evaluation Data	Evaluation Code	Training Code	Model Metadata (Optional)	Research Paper	Datasets	Data Preprocessing Code	Model Parameters (Intermediate)
Aquila-VL-2B		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
FinGPT-mt_llama3-8b_lora		~	~	~	~	~	~	~	✓	✓	✓	✓	✓	~	~	~	✓	✗
Mixtral-8x7B		~	✓	~	✓	~	~	✗	✓	✓	✓	✗	✗	~	✓	✗	✗	✗
Gemma-7B		~	✓	~	✓	~	~	✗	✓	✓	✓	✗	✗	~	✓	✗	✗	✗
DeepSeek-R1		~	~	~	✓	~	~	✗	✓	✓	✓	✗	✗	~	~	✗	✗	~
DeepSeek-V3-0324		~	~	~	✓	~	~	✗	✓	✓	✓	✗	✗	~	~	✗	✗	~
Qwen2.5-14B		~	~	~	✓	~	~	✗	✓	✓	✓	✗	✗	~	~	✗	✗	✗
DeepSeek-V3		~	~	~	✓	~	~	✗	✓	✓	✓	✗	✗	~	~	✗	✗	~
Granite-3.1-8B-Instruct		~	~	~	✓	~	~	✗	✓	✓	✓	✗	✗	~	✗	✗	✗	✗
Qwen2.5-72B		~	~	~	~	~	~	✗	~	~	✓	✗	✗	~	~	✗	✗	✗
Llama-3.1-70B		~	~	~	~	~	~	✗	~	~	✓	✗	✗	~	~	✗	✗	✗
Mixtral-8x22B		~	~	~	✗	~	~	✗	✓	✓	✓	✗	✗	~	✗	✗	✗	✗
GPT-4o		~	✓	~	✗	✗	✗	✗	✗	✗	✓	✗	✗	~	✓	✗	✗	✗
Gemini-2.0		~	~	~	✗	✗	✗	✗	✗	✗	✓	✗	✗	~	~	✗	✗	✗
Claude 3		~	~	~	✗	✗	✗	✗	✗	✗	✓	✗	✗	~	~	✗	✗	✗
o3-mini		~	~	~	✗	✗	✗	✗	✗	✗	✓	✗	✗	~	✗	✗	✗	✗
ChatGPT		~	~	~	✗	✗	✗	✗	✗	✗	✓	✗	✗	~	✗	✗	✗	✗
BloombergGPT		~	~	~	✗	✗	✗	✗	✗	✗	✗	✗	✗	~	~	✗	✗	✗

Figure 3: An evaluation of models' openness under the MOF. ✓ means 'released under an acceptable open license'; ∼ means 'optional or released but not under an acceptable open license'; ✗ means 'not released'. The fine-tuned model, FinGPT-mt_llama3-8b_lora, is evaluated only for the adapter.

4.1 Case Study of Recent GenAI Models

We evaluate DeepSeek V3 [12] and R1 [11] using the MOF as a case study. The process is as follows: 1). List all artifacts released for the DeepSeek-V3/R1 model, identifying their names, locations, versions, and licenses; 2). Map the artifacts to the 16 components; 3). For each MOF component present, check if it uses an acceptable open license from Fig. 2; 4). Check the components against the list for the 3 classes in Fig. 1. Classify the model at the highest tier where all required components in the class employ open licenses; 5). Create the MOF.JSON file, including all required details in Step 1; and 6). Assert the MOF class using the MOT.

The evaluation results show that DeepSeek-V3 [12] and R1 [11] are progressing to the Class III Open Model, as shown in Fig. 3. DeepSeek-R1 [11] releases both code and model parameters under the MIT license. DeepSeek-V3 [12] has code components under the MIT license and model parameters under the DeepSeek License Agreement. The DeepSeek License Agreement is derived from the BigScience OpenRAIL-M license. It grants copyright and patent for the reproduction, modification, and distribution of the model. However, it imposes restrictions on illegal, military, and unethical usage. Therefore, the DeepSeek License Agreement is not considered an open license.

Following the efforts of promoting the MOF, the new version, DeepSeek-V3-0324, has model parameters released under the MIT license. Though MIT is an inappropriate open-source license for model parameters as data, it is still a meaningful step towards openness.

4.2 Hybrid Releases

Openness has always been a binary decision in the open-source movement; software is either open-source or not, with no in-between[1]. A developer either released their software under an OSI-approved license or they did not. If any essential component was not released under an open-source license, the entire release was no longer considered open source. The MOF follows this principle. When any component is not released using an open license as described in Fig. 1, that component is not deemed open and does not qualify for an MOF class. Removing a component that moves the project into a lesser class is acceptable if all remaining components are released with open licenses.

To qualify as a Class III project, the model, its parameters, and a technical report that describes the work, along with evaluation results and model and data cards, must be released with open licenses. If not, the project cannot be considered open. This includes projects that use modified open licenses and implement restrictions or acceptable uses.

It should be noted that the MOF classifies models and their components on completeness when they are open. The reader should not confuse the classification system with being a gradient measure of openness [60], but rather a measurement of the completeness of a release in adherence with open science principles [52, 9, 69].

4.3 OpenMDW License

The Linux Foundation develops the Open Model, Data, and Weights License Agreement V1.0 (OpenMDW V1.0) to address the licensing challenges. The OpenMDW V1.0 is the first permissive license for ML models and their associated artifacts, i.e., Model Materials. Its development originated from the MOF. It aims to provide a single, permissive license agreement that ensures consistency and clarity across all components of an open AI model release. It has the following features: 1). This license grants permission to use, modify, and distribute without restrictions under all relevant intellectual property regimes, including copyright, patent, database, and trade secret rights; 2). Outputs generated by using the Model Materials are not subject to restrictions or obligations; and 3). It has an attribution requirement where users need to include a copy of this license and additional notices for redistribution. This license simplifies the adoption of the MOF, facilitates ML model sharing, and helps address legal ambiguities around model artifacts and outputs.

4.4 Model Openness Tool

The Model Openness Tool (MOT) complements the Model Openness Framework (MOF). It provides a practical, user-friendly way for model producers to apply the MOF framework. It ensures clarity on the permissible uses and restrictions of the model and its various parts. The MOT enables users to 1) comprehend the completeness and openness of ML models in the MOT catalog, 2) evaluate the openness of their own models based on released components and associated licenses, and 3) submit models to the MOT catalog. The evaluated model receives an openness score and badge based on the degree to which each criterion is fulfilled. By offering a practical and user-friendly mechanism, the MOT facilitates the application of the MOF. The MOF badges are being adopted by different open model leaderboards, such as Open Financial LLM Leaderboard [34].

The MOT currently hosts the evaluations of 235 models, providing details of their MOF classes and the licenses of the components. Fig. 3 shows the evaluation results of some open and closed models. Closed models, such as GPT-4o [48], are accurately measured as close due to limited disclosures. Most models that claimed to be open, such as DeepSeek-V3 [12] and Llama [19], are progressing towards the Class III Open Model. However, their model parameters are released under inappropriate open-source licenses or restrictive licenses. The documentation of these models, such as model cards, is often unlicensed. Aquila-VL-2B [20] achieves Class I Open Science, with all models released under open licenses. The MOF provides model producers with a practical roadmap towards higher MOF classes for greater transparency and better alignment with the principles of openness.

5 Conclusion

The MOF provides a clear methodology for evaluating and enhancing the openness and completeness of ML models. It outlines specific components that should be openly released, including training data, code, model architecture, model parameters, and documentation, among others, as well as with which licenses. This framework gives model producers a roadmap to follow for reproducible and transparent AI development. The MOT provides a practical, user-friendly way to apply the MOF framework. To address the licensing challenges widespread among ML models, the OpenMDW License Agreement is developed based on the MOF, covering the ML model and associated artifacts in a single permissive license.

The widespread adoption of the MOF promises to establish completeness and openness as core tenets of responsible AI, ultimately promoting a more transparent and trustworthy advancement of AI R&D. We encourage the wider AI community to recognize and reward the complete and open distribution of models. With carefully designed incentives, policies, and community norms, open source and open science ideals can become the norm in AI R&D, rather than the exception.

References

[1] AI Security Institute. Inspect: A framework for large language model evaluations. *AI Security Institute*, 2024.

[2] Vijay Arya, Rachel K. E. Bellamy, Pin-Yu Chen, Amit Dhurandhar, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Q. Vera Liao, Ronny Luss, Aleksandra Mojsilović, Sami Mourad, Pablo Pedemonte, Ramya Raghavendra, John Richards, Prasanna Sattigeri, Karthikeyan Shanmugam, Moninder Singh, Kush R. Varshney, Dennis Wei, and Yunfeng Zhang. One explanation does not fit all: A toolkit and taxonomy of AI explainability techniques. *arXiv:1909.03012*, 2019.

[3] Adrien Basdevant, Camille François, Victor Storchan, Kevin Bankston, Ayah Bdeir, Brian Behlendorf, Merouane Debbah, Sayash Kapoor, Yann LeCun, Mark Surman, Helen King-Turvey, Nathan Lambert, Stefano Maffulli, Nik Marda, Govind Shivkumar, and Justine Tunney. Towards a framework for openness in foundation models: Proceedings from the columbia convening on openness in artificial intelligence. *arXiv:2405.15802*, 2024.

[4] Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623. Association for Computing Machinery, 2021.

[5] Abeba Birhane, Ryan Steed, Victor Ojewale, Briana Vecchione, and Inioluwa Deborah Raji. Ai auditing: The broken bus on the road to ai accountability. In *IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 612–643. IEEE, 2024.

[6] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, volume 81 of *Proceedings of Machine Learning Research*, pages 77–91, 2018.

[7] Joel Castaño, Silverio Martínez-Fernández, and Xavier Franch. Lessons learned from mining the hugging face repository. In *Proceedings of the 1st IEEE/ACM International Workshop on Methodological Issues with Empirical Studies in Software Engineering*, pages 1–6, 2024.

[8] Henry Chesbrough. *Open Innovation: The New Imperative for Creating and Profiting from Technology*. Harvard Business Press, 2003.

[9] Henry Chesbrough. From open science to open innovation. *ESADE*, 2015.

[10] Data & Trust Alliance. Data provenance standards. *Data & Trust Alliance*, 2024.

[11] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, et al. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv:2501.12948*, 2025.

[12] DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, et al. DeepSeek-V3 technical report. *arXiv:2412.19437*, 2025.

[13] Francisco Eiras, Aleksander Petrov, Bertie Vidgen, Christian Schroeder, Fabio Pizzati, Katherine Elkins, Supratik Mukhopadhyay, Adel Bibi, Aaron Purewal, Csaba Botos, Fabro Steibel, Fazel Keshtkar, Fazl Barez, Genevieve Smith, Gianluca Guadagni, Jon Chun, Jordi Cabot, Joseph Imperial, Juan Arturo Nolazco, Lori Landay, Matthew Jackson, Phillip H. S. Torr, Trevor Darrell, Yong Lee, and Jakob Foerster. Risks and opportunities of open-source generative AI. *arXiv:2405.08597*, 2024.

[14] Ellen Enkel, Oliver Gassmann, and Henry Chesbrough. Open R&D and open innovation: Exploring the phenomenon. *R&D Management*, 39(4):311–316, 2009.

[15] European Union. Regulation (eu) 2024/1689 of the european parliament and of the council of 13 june 2024 laying down harmonised rules on artificial intelligence and amending regulations (ec) no 300/2008, (eu) no 167/2013, (eu) no 168/2013, (eu) 2018/858, (eu) 2018/1139 and (eu) 2019/2144 and directives 2014/90/eu, (eu) 2016/797 and (eu) 2020/1828 (artificial intelligence act). *Official Journal of the European Union*, 2024.

[16] Francisco J García-Peñalvo, Carlos Garcia de Figuerola, and José A Merlo. Open knowledge: Challenges and facts. *Online Information Review*, 34(4):520–539, 2010.

[17] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92, 2021.

[18] Andrés Guadamuz González. Open science: Open source licenses in scientific research. *North Carolina Journal of Law & Technology*, 7:321, 2006.

[19] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, et al. The Llama 3 herd of models. *arXiv:2407.21783*, 2024.

[20] Shuhao Gu, Jialing Zhang, Siyuan Zhou, Kevin Yu, Zhaohu Xing, Liangdong Wang, Zhou Cao, Jintao Jia, Zhuoyi Zhang, Yixuan Wang, Zhenchong Hu, Bo-Wen Zhang, Jijie Li, Dong Liang, Yingli Zhao, Yulong Ao, Yaoqi Liu, Fangxiang Feng, and Guang Liu. Infinity-MM: Scaling multimodal performance with large-scale and high-quality instruction data. *arXiv:2410.18558*, 2024.

[21] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM Computing Surveys (CSUR)*, 51(5):1–42, 2018.

[22] Tianyu Han, Lisa C Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K Bressem. MedAlpaca–An open-source collection of medical conversational AI models and training data. *arXiv:2304.08247*, 2023.

[23] Philip Heltweg and Dirk Riehle. A systematic analysis of problems in open collaborative data engineering. *ACM Transactions on Social Computing*, 6(3-4):1–30, 2023.

[24] Matthew Hutson. AI researchers allege that machine learning is alchemy. *Science*, 360(6388):861, 2018.

[25] Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts. *arXiv:2401.04088*, 2024.

[26] Sayash Kapoor, Rishi Bommasani, Kevin Klyman, Shayne Longpre, Ashwin Ramaswami, Peter Cihon, Aspen Hopkins, Kevin Bankston, Stella Biderman, Miranda Bogen, Rumman Chowdhury, Alex Engler, Peter Henderson, Yacine Jernite, Seth Lazar, Stefano Maffulli, Alondra Nelson, Joelle Pineau, Aviya Skowron, Dawn Song, Victor Storchan, Daniel Zhang, Daniel E. Ho, Percy Liang, and Arvind Narayanan. On the societal impact of open foundation models. *arXiv:2403.07918*, 2024.

[27] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *International Conference on Machine Learning*, pages 17061–17084. PMLR, 2023.

[28] Rob Kitchin. *The Data Revolution: Big Data, Open Data, Data Infrastructures and Their Consequences*. Sage, 2014.

[29] Max Langenkamp and Daniel N. Yue. How open source machine learning software shapes AI. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, AIES '22, page 385–395, New York, NY, USA, 2022. Association for Computing Machinery.

[30] Harry Law and Sébastien Krier. Open-source provisions for large models in the AI Act. *Cambridge Journal of Science and Policy*, 2023.

[31] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[32] Andreas Liesenfeld and Mark Dingemanse. Rethinking open source generative AI: Open-washing and the EU AI Act. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency*. Association for Computing Machinery, 2024.

[33] Andreas Liesenfeld, Alianda Lopez, and Mark Dingemanse. Opening up ChatGPT: Tracking openness, transparency, and accountability in instruction-tuned text generators. In *International Conference on Conversational User Interfaces*, pages 1–6, 2023.

[34] Shengyuan Colin Lin, Felix Tian, Keyi Wang, Xingjian Zhao, Jimin Huang, Qianqian Xie, Luca Borella, Matt White, Christina Dan Wang, Kairong Xiao, Xiao-Yang Liu Yanglet, and Li Deng. Open FinLLM Leaderboard: Towards financial AI readiness, 2025.

[35] Yi-Hsuan Lin, Tung-Mei Ko, Tyng-Ruey Chuang, Kwei-Jay Lin, et al. Open source licenses and the creative commons framework: License selection and comparison. *Journal of Information Science and Engineering*, 22:1:1–17, 2006.

[36] Jordan Maris. Meta's LLaMa license is still not open source. *Open Source Initiative*, 2025.

[37] Timothy R McIntosh, Teo Susnjak, Nalin Arachchilage, Tong Liu, Dan Xu, Paul Watters, and Malka N Halgamuge. Inadequacies of large language model benchmarks in the era of generative artificial intelligence. *IEEE Transactions on Artificial Intelligence*, 2025.

[38] Filippo Menczer, David Crandall, Yong-Yeol Ahn, and Apu Kapadia. Addressing the harms of AI-generated inauthentic content. *Nature Machine Intelligence*, 5(7):679–680, 2023.

[39] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency*, page 220–229, New York, NY, USA, 2019. Association for Computing Machinery.

[40] Jakob Mökander, Jonas Schuett, Hannah Rose Kirk, and Luciano Floridi. Auditing large language models: A three-layered approach. *AI and Ethics*, pages 1–31, 2023.

[41] Jennifer C Molloy. The open knowledge foundation: Open data means better science. *PLoS Biology*, 9(12):e1001195, 2011.

[42] Peter Murray-Rust. Open data in science. *Nature Precedings*, pages 1–1, 2008.

[43] Micah Musser. A cost analysis of generative language models and influence operations. *arXiv:2308.03740*, 2023.

[44] Michael Nolan. Llama and ChatGPT are not open-source. *IEEE Spectrum*, 2023.

[45] B. A. Nosek, G. Alter, G. C. Banks, D. Borsboom, S. D. Bowman, S. J. Breckler, S. Buck, C. D. Chambers, G. Chin, G. Christensen, M. Contestabile, A. Dafoe, E. Eich, J. Freese, R. Glennerster, D. Goroff, D. P. Green, B. Hesse, M. Humphreys, J. Ishiyama, D. Karlan, A. Kraut, A. Lupia, P. Mabry, T. Madon, N. Malhotra, E. Mayo-Wilson, M. McNutt, E. Miguel, E. Levy Paluck, U. Simonsohn, C. Soderberg, B. A. Spellman, J. Turitto, G. VandenBos, S. Vazire, E. J. Wagenmakers, R. Wilson, and T. Yarkoni. Promoting an open research culture. *Science*, 348(6242):1422–1425, 2015.

[46] NTIA. Ntia kicks off public engagement on executive order ai work. *National Telecommunications and Information Administration*, 2023.

[47] Open Source Initiative. The open source AI definition – 1.0. *Open Source Initiative*, 2024.

[48] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, et al. GPT-4 technical report. *arXiv:2303.08774*, 2024.

[49] Cailean Osborne. Public-private funding models in open source software development: A case study on scikit-learn. *arXiv:2404.06484*, 2024.

[50] Cailean Osborne, Jennifer Ding, and Hannah Rose Kirk. The AI community building the future? A quantitative analysis of development activity on Hugging Face Hub. *Journal of Computational Social Science*, pages 1–39, 2024.

[51] Michael A Peters and Peter Roberts. *Virtues of Openness: Education, Science, and Scholarship in the Digital Age*. Routledge, 2015.

[52] Jason Phang, Herbie Bradley, Leo Gao, Louis Castricato, and Stella Biderman. EleutherAI: Going beyond "open science" to "science in the open". *arXiv:2210.06413*, 2022.

[53] Joelle Pineau, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Larivière, Alina Beygelzimer, Florence d'Alché Buc, Emily Fox, and Hugo Larochelle. Improving reproducibility in machine learning research (a report from the neurips 2019 reproducibility program). *Journal of Machine Learning Research*, 22(164):1–20, 2021.

[54] Kunat Pipatanakul, Phatrasek Jirabovonvisut, Potsawee Manakul, Sittipong Sripaisarnmongkol, Ruangsak Patomwong, Pathomporn Chokchainant, and Kasima Tharnpipitchai. Typhoon: Thai large language models. *arXiv:2312.13951*, 2023.

[55] Inioluwa Deborah Raji and Joy Buolamwini. Actionable auditing: Investigating the impact of publicly naming biased performance results of commercial AI products. In *AAAI/ACM Conference on AI, Ethics, and Society*, pages 429–435, 2019.

[56] Inioluwa Deborah Raji, Andrew Smart, Rebecca N White, Margaret Mitchell, Timnit Gebru, Ben Hutchinson, Jamila Smith-Loud, Daniel Theron, and Parker Barnes. Closing the AI accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency*, pages 33–44. Association for Computing Machinery, 2020.

[57] Brianna Richardson and Juan E Gilbert. A framework for fairness: A systematic review of existing fair AI solutions. *arXiv:2112.05700*, 2021.

[58] Jonas Schuett, Ann-Katrin Reuel, and Alexis Carlier. How to design an AI ethics board. *AI and Ethics*, pages 1–19, 2024.

[59] Elizabeth Seger, Noemi Dreksler, Richard Moulange, Emily Dardaman, Jonas Schuett, K Wei, Christoph Winter, Mackenzie Arnold, Seán Ó hÉigeartaigh, Anton Korinek, et al. Open-sourcing highly capable foundation models: An evaluation of risks, benefits, and alternative methods for pursuing open-source objectives. *arXiv:2311.09227*, 2023.

[60] Irene Solaiman. The gradient of generative AI release: Methods and considerations. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency*, pages 111–122. Association for Computing Machinery, 2023.

[61] Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, Gretchen Krueger, Jong Wook Kim, Sarah Kreps, et al. Release strategies and the social impacts of language models. *arXiv:1908.09203*, 2019.

[62] Peter Suber. *Open Access*. The MIT Press, 2012.

[63] Jonathan P Tennant, François Waldner, Damien C Jacques, Paola Masuzzo, Lauren B Collister, and Chris HJ Hartgerink. The academic, economic and societal impacts of open access: An evidence-based review. *F1000Research*, 5, 2016.

[64] David Thiel, Melissa Stroebel, and Rebecca Portnoff. Generative ML and CSAM: implications and mitigations. *Stanford Internet Observatory Cyber Policy Center*, 2023.

[65] Andreas Tsamados, Luciano Floridi, and Mariarosaria Taddeo. The cybersecurity crisis of artificial intelligence: Unrestrained adoption and natural language-based attacks. *arXiv:2311.09224*, 2023.

[66] Ruben Vicente-Saez and Clara Martinez-Fuentes. Open science now: A systematic literature review for an integrated definition. *Journal of Business Research*, 88:428–436, 2018.

[67] Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, et al. Ethical and social risks of harm from language models. *arXiv:2112.04359*, 2021.

[68] David Gray Widder, Sarah West, and Meredith Whittaker. Open (for business): Big tech, concentrated power, and the political economy of open AI. *Concentrated Power, and the Political Economy of Open AI*, 2023.

[69] John Willinsky. The unacknowledged convergence of open source, open access, and open science. *First Monday*, 10(8), 2005.

[70] Irving Wladawsky-Berger. Are open AI models safe? *The Linux Foundation*, 2023.

[71] Xianjun Yang, Xiao Wang, Qi Zhang, Linda Petzold, William Yang Wang, Xun Zhao, and Dahua Lin. Shadow alignment: The ease of subverting safely-aligned language models. *arXiv:2310.02949*, 2023.

A Related Work

A.1 Benefits and Risks of Openness in AI

There has been much debate about the benefits and risks of releasing AI models [61, 59, 26, 60, 30, 13]. On the one hand, open models have many advantages over closed ones. They improve security through distributed development and auditing [70, 55], support adaptability and customization for diverse domains and languages [26, 54], and drive advances in science [71, 27, 22]. On the other hand, the openness of models introduces risks, such as the generation of misinformation [43, 38], illegal content [64], and security vulnerabilities [65].

Open foundational models have five distinctive properties that present *both* benefits and risks: 1) broader access, 2) greater customizability, 3) local adaptation and inference ability, 4) the inability to rescind model access, and 5) the inability to monitor or moderate model usage [26]. A systematic review [13] argues that the benefits of open generative AI models outweigh the risks.

A.2 Lack of Openness in “Open Source” AI

Some ML models with publicly available weights are falsely promoted as “open source”[44, 36, 33]. Such models may more accurately be described as “open-weight models” [32]. This misrepresentation is in part due to the misuse of open-source licenses. They were designed for conventional software code and are not appropriate for the intricacies of ML models[1]. The misrepresentation of models as “open source” by companies has been characterized as “openwashing”[44, 33, 32], where “open” is used imprecisely and loosely to describe both minimally and fully transparent systems [68].

A concerning number of models also have licensing issues for openness. 64.67% of the models and 72.13% of the datasets on Hugging Face Hub are unlicensed [50]. Some models are released under restrictive licenses that do not meet the standards required of open licenses [44, 36]. In addition, some fine-tuned models are released under open-source licenses (e.g., Apache 2.0), even when their base models use restrictive licenses. However, altering the original license is not legally permitted. This creates confusion in the ecosystem and can have legal consequences for model consumers.

Another challenge is that most models fall short in their completeness, only releasing model architectures and final trained parameters. The technical reports and model cards usually provide limited information on the source and treatment of training data, fine-tuning, or alignment methods [48, 25]. Evaluation results often cannot be reproduced independently due to the lack of models’ disclosure [37]. As a result, downstream model consumers have to rely on limited and unverifiable claims reported by the model producers.

These challenges motivate our creation of a ranking system to promote openness and completeness. Model producers should include all artifacts of their work under appropriate open licenses, including datasets and code for training, validation, and evaluation, as well as detailed documentation.

A.3 Evaluating Openness in AI

There is not yet a formally agreed-upon definition of open AI. Broadly, open AI refers to the concept of transparency and accessibility in AI R&D. It requires to share key artifacts along the model development lifecycle, including data, code, models, and publications, under open licenses. These licenses allow free access, inspection, modification, or distribution of models.

New standards are being developed to address the above shortcomings. The Open Source Initiative released the first version of the Open Source AI Definition [47]. It requires an AI system and its discrete components to be available for free use, study, modification, and sharing. The Mozilla Foundation, in collaboration with leading scholars and practitioners, presented a framework to understand openness across the AI stack [3]. The EU AI Act is the first comprehensive regulation of AI by a major regulator [15]. The other standards include tools for auditing model explainability, fairness, and robustness [5, 21, 56, 57, 2, 40]; frameworks to evaluate model openness, such as the AAAI Reproducibility Checklist [45] and the NeurIPS 2019 ML Reproducibility Checklist [53]; the establishment of ethics review boards in AI research labs [58]; as well as work by government agencies, including NIST and NTIA in the USA [46] and the AI Safety Institute in the UK [1].

[1] `https://opensource.org/license`

However, prior approaches do not evaluate both the completeness and openness of models. The MOF reinforces existing approaches by objectively evaluating and classifying models based on which components of the development lifecycle are released under open licenses. The MOF encourages model producers to strive for complete transparency and usability without restrictions.

B Understanding the Concepts and Culture of Openness and Completeness

Before presenting the details of the MOF, we review the concepts of openness and completeness in science and technology. They enable transparency, reproducibility, and collaboration in research, facilitating the democratization of AI.

For simplicity, this paper refers to any person or entity that develops and trains a first-generation model as a "model producer" or simply a "producer". Any person or entity that adopts, consumes, alters, or uses a model and corresponding artifacts for any purpose including modifying weights through fine-tuning is referred to as a "model consumer" or simply a "consumer". We also use the terms "ML" and "ML model" to broadly describe any model, whether classical machine learning or deep learning, and both generative and discriminative.

B.1 Openness

Openness is the practice of freely sharing the methodology, progress, and products of R&D with the public without restrictions on access, inspection, modification, or distribution [51]. It supports reproducibility, accountability, and cumulative innovation by enabling research and developer communities to review, discuss, reuse, and extend upon prior work [66]. The release of materials should be under permissive open licenses tailored to the type of content. The MOF aligns with wider open science principles and the vision of open AI that requires more than open-source licenses for code components for models to be considered open. For example, non-code elements like datasets and research papers need an appropriate license that suits their format, such as open-data or open-content licenses, which are not currently OSI-approved licenses.

B.2 Completeness

Completeness is a core tenet of open science [66]. We define completeness as the availability of key artifacts produced during the full lifecycle of conducting research or the engineering of a technical product, enabling comprehensive transparency, inspection, evaluation, and reproducibility. In the context of ML, completeness entails releasing all the key components associated with developing an ML model rather than just selected artifacts. It empowers unfettered scrutiny into model genetics: curation and treatment of training data, feature engineering, neural architectures, weight evolution, training configurations, model performance across diverse benchmarks, replication of model producer claims, and other byproducts of the model development lifecycle. The MOF promotes full completeness by defining an ascending hierarchy of criteria for releasing key artifacts, encouraging model producers to release all artifacts involved in the model development lifecycle.

We distinguish completeness from openness to avoid confusion. "Openness" has unfortunately become a vague and confusing term [68, 33], packed with multiple definitions, uses, or dimensions, such as the licensing, availability, or thoroughness of artifacts. For instance, a model producer may claim that their model is "open" but model consumers may not know if it is open because it employs open licenses, because it is made publicly available, because it provides additional components like datasets, or because the components released are thorough or usable. For this reason, we use the term "completeness" to measure the availability of components that are released with models (with the goal of full completeness) and the term "openness" to describe the usage of permissive licenses for components.

B.3 Open Licenses

Open licenses are legal mechanisms that allow content and artifacts to be freely accessed, used, modified, and shared under permissive terms. They are essential for operationalizing openness. Different licenses have emerged for addressing rights, responsibilities, and permissible usage for data, publications, code, and other research outputs. Open licenses solve key problems with closed, restricted systems, including:

- Enabling free access without paywalls or subscriptions.
- Allowing reproduction, analysis, and extension of work.
- Disseminating contributions back to the community.
- Progressing cumulatively by building on prior ideas.
- Fostering collaboration across organizational and geographic boundaries.
- Promoting transparency and accountability.
- Mitigating anti-competitive behavior or rent-seeking.

For research papers and scholarly works, Creative Commons (CC) licenses are widely adopted, which allow free distribution and reuse with conditions, such as requiring attribution and allowing commercial use and derivative works. Common choices for open licenses are CC BY (attribution) and CC BY-SA (Attribution-ShareAlike). Using permissive CC licenses for papers, technical reports, and documentation provides rights to reproduce, expand, and translate the works [35].

For software code, many open-source licenses have been developed. The Open Source Definition and the list of approved open-source licenses are maintained by the OSI[1]. Prominent examples include the MIT, Apache 2.0, and the 3-Clause BSD license, which allow inspection, modification, and redistribution of code while requiring preservation of copyright and license terms. Alternative licenses, such as the Llama 2 license, OpenRAIL, and AI2 ImpACT licenses, are not considered open-source licenses due to their restrictions on usage [36].

For datasets, typical licenses are Creative Commons licenses like Creative Commons Zero (CC0), CC BY, and CC BY-SA, as well as the Community Data License Agreement (CDLA-Permissive) and the Open Data Commons licenses, such as Public Domain Dedication and License (PDDL) and the Open Data Commons Attribution License License (ODC-By). They provide terms for sharing data openly while addressing concerns regarding attribution, permissive usage, and liability [35].

C Understanding the Domains in Openness and Completeness

C.1 Open Knowledge

Open knowledge is an overarching philosophy and larger movement that encompasses all the preceding areas of openness, revolving around the free and public sharing of information and insights across various domains [16, 41]. This entails making knowledge resources accessible to everyone and contributing to a wider pool of shared understanding. Open knowledge practices also involve ensuring that the information is ethically curated and disseminated, upholding principles of integrity and respect for intellectual property. The Wikimedia Foundation, Open Knowledge Foundation, and Science Commons are leading organizations in the open knowledge community.

C.2 Open Science

Open science refers to the practice of making all stages of the scientific process transparent and accessible to others [66, 9]. This includes publishing research papers, data, source code, code notebooks, and any information or tools needed to replicate research. The goals of open science are to enable reproducibility, collaboration, and advance scientific research building on previous knowledge [66]. Open science in AI is the gold standard for ensuring reproducibility and transparency. However, much of the training data, model details, and code of SOTA AI systems remain proprietary. This limits reproducibility, hinders research, and increases concerns around bias and safety. The MOF aims to promote the spirit and methodology of open science in the AI R&D community.

C.3 Open Access

Open access is the process of making research outputs like publications freely available to read without subscriptions or paywalls, enabling broad dissemination of knowledge. [62, 63]. There are various open-access platforms like Cornell University's arXiv, which make publications, often distributed under an open license, freely available for review. Furthermore, the adoption of open access policies, mandates, and licenses by journals and conferences have contributed to greater access to research. Before open access, research publications were mostly locked behind expensive journal subscriptions and paywalls, which limited the discoverability and use of knowledge. The open access movement has made more research freely available to all. Open access speeds the dissemination of discoveries to scientists and the public, and it facilitates reproducibility and meta research. As a result, entry barriers to accessing research have greatly reduced and public access to AI research papers has helped advance the field, including many of the developments and enhancements to the transformer architecture that powers the latest highly-capable LLMs.

C.4 Open Collaboration and Open Community

Open collaboration encourages cooperative efforts across institutions, disciplines, and borders, involving more inclusive and diverse participation in the development of science and technology [14, 9, 8]. Open community goes beyond open collaboration, and it concerns the creation and sustainability of a shared community with neutral governance, where projects can be worked on collaboratively in an equitable environment that embraces principles of openness. The LF AI & Data and Generative AI Commons are examples of open communities.

C.5 Open Source Software

Open source software (OSS) involves publishing software code under licenses that grant users independence and control over the technology by allowing inspection, modification, and redistribution of the code without restrictions[2]. OSI-approved licenses like Apache 2.0 and MIT have been key to enabling worldwide collaborative development, freedom of choice, and accelerated progress [18]. OSS has emerged as an indispensable component of AI R&D [29, 49].

C.6 Source Available

Source available should not be confused with open source. Source available originated from conventional software development, where a developer provides access to the source code, but the licenses are not open-source. This means they include restrictions that consumers must fully understand before agreeing to use it. Some have referred to these projects as open access, but this is a misnomer since open access applies to documentation without paywalls. Most open-washed projects are examples of source available due to their restrictive licensing [44, 36].

C.7 Open Data

Open data refers to the public release of datasets, databases, and other structured data used for research, enabling access and reuse [42, 28]. This practice upholds scientific reproducibility, allows reanalysis, and spurs innovation [23]. Open content, on the other hand, refers to the sharing of creative materials and unstructured data. Both open-data and open-content licenses exist, with open-data licenses often applicable to both data and content. Open data emphasizes the standardization of datasets, addressing transparency and requiring comprehensive descriptions of data collection methods and assessments for intrinsic bias. Furthermore, accessibility is a cornerstone of open data, with datasets expected to be readily available without personal requests or paywalls, promoting transparency and enabling scrutiny. In the context of AI R&D, the Datasets and Benchmarks track at NeurIPS underscores the paramount importance of openly releasing machine learning datasets [17].

D MOF Process

D.1 MOF Process Overview

Unlike other frameworks that attempt to dictate how model producers should build and train their models or create a release path on how models should be released, we take a more objective approach by evaluating models based on their completeness and openness. This approach does not constrain model producers into a single methodology but rather lays out a pliable process that acts as a guideline

[2] `https://opensource.org/license`

to help model producers create the most complete and open models. At the completion of the process, model producers receive a badge for their MOF class that clearly demonstrates to the public their commitment to both completeness and openness.

The MOF process generally follows these steps:

1. Inventory of artifacts
   (a) Comprehensively list all artifacts involved in creating the model (data, code, documentation, etc).
   (b) Capture details like component names, component locations, versions and licenses.
2. Map to MOF components
   (a) Align inventory items to the 16 components.
   (b) Multiple inventory elements may map to a single standard component.
3. Verify licenses
   (a) For each MOF component present, check if it uses an acceptable open license from Fig. 2.
   (b) If licenses are incompatible, the model cannot be classified.
4. Determine completeness
   (a) Check inventory against the component list for the 3 classes in Fig. 1.
   (b) Classify model at the highest tier where all required components in the class employ open licenses.
   (c) Model meets Class III at a minimum when using open licenses.
5. Generate MOF.JSON
   (a) Create the MOF.JSON file, either using the Model Openness Tool (MOT) or manual means.
   (b) Include all artifacts, licenses, locations and other required data to meet the MOF requirements.
6. Self-assert classification
   (a) With inventory, mapping, and MOF.JSON file finalized, the model producer asserts the appropriate class using the Model Openness Tool (MOT) or through self-assessment.
   (b) The model producer must stand behind their completeness and openness claims.
7. Badging and validation
   (a) The model producer uses the MOT for badging classified models.
   (b) MOT provides the MOF.JSON file and badge code for inclusion with project files.
   (c) Community helps ensure accurate labeling by filing disputes.

This process determines a model's location on the spectrum, guiding model producers in improving openness and consumers in evaluating fitness of models for their usage.

D.2 Preparing the Distribution

All projects must include a LICENSE file that describes the licenses used for the project. Conventionally a LICENSE file would include a single license, however it is recommended that the LICENSE file include all licenses that apply to the project. For instance if software is covered under Apache 2.0 and all documentation and data use CC-BY-4.0, then the text of both licenses should be included in the LICENSE file in their entirety including the license heading in order to distinguish what text belongs to which license. Alternatively, a distribution can contain different LICENSE files that are bound to the different components included in the distribution. Ideally the LICENSE files for each component should be located in the base directory of the component that they cover. The MOF.JSON file records the path to the appropriate LICENSE file for each component included in the distribution and facilitates both the per component LICENSE method and the single LICENSE file method.

In addition to the LICENSE file, the distribution must include an MOF.JSON file providing details about the MOF version, release details, included components, and their licenses. This file can be generated with the MOT maintained by the Generative AI Commons or created manually or automatically. It is important to note that when a component is not released with the distribution, it should not appear in the MOF.JSON file. When a component is released but does not use an open license or it uses a custom license, it should not be included in the MOF.JSON file either. The MOF.JSON file only references components that are released using an open license.

D.3 MOF.JSON Structure

The MOF JSON file is structured as a single MOF object defined at the root of the JSON file (see GitHub[3]. Specifically, under the root there are three required, nested objects with their own set of variables:

- **Framework**: This object contains the details related to the framework itself, including the following required variables:
  - **name**: The name of the framework. The variable type is string.
  - **version**: The version number of the framework. The variable type is string.
  - **date**: The publication date of the framework. The variable type is string in YYYY-MM-DD format.
- **Release**: This object contains the details of the model being released. There are a number of variables:
  - **name**: The name of the release. The variable type is string.
  - **version**: The version of the release, which can be the parameter count or another identifier that distinguishes the model from previous versions and versions of the same model with different parameter counts. The variable type is string.
  - **date**: The date of the release. The variable type is string in format "YYYY-MM-DD".
  - **type**: The nature of the model, i.e., language model, image generation, audio generation, image classification, statistical ML, or any number of other types of models. The variable type is string.
  - **architecture**: The model architecture employed, i.e., transformer, diffusion, GAN, NERF, VGG, Resnet, K-means, or any other type of model architecture. The variable type is string.
  - **treatment**: Any type of post-training treatment, like fine-tuning, constitutional alignment, RLHF or any other treatment that otherwise modifies the parameters of the original model. If no treatment has been applied then this variable is an empty string. The variable type is string.
  - **origin**: The original model, generally this is the foundation model. If this is not a foundation model in the release, then this variable contains the name and version of the model that was modified. The variable type is string or left empty for foundation or non-derivative models.
  - **producer**: The name model producer or publisher, could be a company, organization, group or individual. The variable type is string.
  - **contact**: The email address for the model producer or publisher. The variable type is string.
  - **mof_class**: The qualifying MOF class of the release as generated by the Model Openness Checker. The variable type is integer.
- **Components**: This object contains a list of components that are included with the model distribution, as well as each component's details:
  - **description**: A text description of the component. Using the default values is acceptable. When introducing a new component beyond the standard components, include a description of the component.
  - **location**: The location of the component within the distribution, full path is required in UNIX format with leading slash for the root directory. The variable type is string.

[3] `https://github.com/isitopenai/MOF/blob/main/MOF.json`

- **license**: The SPDX identifier of the license(s) used for the component. If multiple licenses are used for a single component, often the case for libraries and tools, they must be provided in a comma-separated list. The value must use a valid SPDX license identifier[4]. The variable type is string.
- **license_path**: The location of the LICENSE file for the component within the distribution, full path is required in POSIX format with leading slash for the root directory. More than one component can point to the same LICENSE file. In the event the component employs multiple licenses, the LICENSE file should contain the text for all the licenses used. Alternatively, multiple license files may be specified, each separated by a comma. However they must correspond in order to the comma separated list of license names provided in the license variable. The variable type is string.

D.4 Class Assignment

The MOF relies on self-reporting and projects are not classified by a central authority. LF AI & Data Generative AI Commons provides a web interface, the MOT, that allows model producers to fill out a web form with the details of their project and in turn the MOT informs the user how their project lines up with the classes in the MOF.

D.5 Badging System

The MOF is designed to be both informational and actionable. As such the Generative AI Commons is implementing a badging program, similar to the OpenSSF Best Practices Badge Program [5]. The badging system is a part of the MOT, and is a free service that allows model producers to perform the following:

- Perform a check the completeness and openness of their model distribution and display which MOF class their model meets
- Receive recommendations on which licenses to use for which components
- Generate an MOF.JSON file for their distribution
- Be provided with code to insert into their README.md file in their Github repository
- Track their model's ranking amongst other models on the MOF scoreboard

For model consumers, they can do the following:

- View the MOF scoreboard to see which models are the most complete and open
- Drill down into model distributions to see which ones meet their completeness and openness requirements
- Quickly see which MOF class a model has attained in the project's Github repo
- Validate that a model has attained an MOF class
- Submit a dispute if they believe that a model is being misrepresented as complete or open

It is incumbent upon the producer of an ML model and its components to accurately include the results of either the MOT or accurately identify the components and licenses included in the distribution in the MOF.JSON file and specify the class the project qualifies for. Misrepresentations will only harm the reputation of the model producer.

E Functionalities of the Model Openness Tool

E.1 View Models

The MOT catalog interface (see Figure 4) presents a tabular view of registered ML models. Each row represents a distinct model, with columns providing key information at a glance. The model

[4] `https://spdx.org/licenses/`

[5] `https://www.bestpractices.dev/en`

name column includes clickable icons that link to the model's repositories on GitHub and Hugging Face Hub, facilitating immediate access to the model's source. The classification and badge are dynamically generated based on the released components and their associated licenses. Upon selecting a model, users are directed to a detailed model page (see Figure 5), which provides:

- A comprehensive overview of the model's MOF classification.
- A component-wise breakdown, categorizing each into released with valid licenses, released with invalid licenses, or unreleased.
- A copyable MOF badge for external use (e.g., in repositories).
- A reporting mechanism for data corrections or updates.

E.2 Evaluate Models

The evaluation interface (see Figure 6) allows users to assess the completeness and openness both self-developed and unregistered models. The process involves:

- Input of license information for each of the 16 MOF components via a dropdown menu.
- Automatic classification of components with empty license fields as unreleased.

Post-evaluation, the MOT generates:

- A model page with a MOF classification score (1-3).
- A component-wise breakdown (as in the catalog model pages).

This score provides a quantitative measure that facilitates easy interpretation of a model's alignment with the principles of openness and objective comparisons between models.

E.3 Submit Model

The submission interface (see Figure 7) guides users through a structured process to add models to the MOT catalog. Key steps include:

- Input of model metadata, including:
  - Name
  - Description
  - Version/parameters
  - Organization
  - Type (e.g., language model, image model, code model)
  - Version/parameter count
  - Architecture (e.g., transformer, diffusion, RNN, CNN, etc.)
  - Treatment (e.g., pre-trained, instruct fine-tuned, or chat fine-tuned)
  - Base model
  - Hugging Face Hub link (if applicable)
- License specification for each of the 16 MOF components via dropdown menus.

Upon submission, the MOT:

- Calculates the MOF classification score (1-3).
- Generates a model page.
- Integrates the model into the public MOT catalog.

This streamlined process ensures consistency in model representation and facilitates the expansion of the MOT database.

E.4 Disputes

The MOF relies on the honesty and transparency of researchers and developers to accurately classify models and to state which components with which licenses they include. Therefore, we also rely on the community to identify projects that have been misrepresented as open and notify the organization that hosts the project about their concerns.

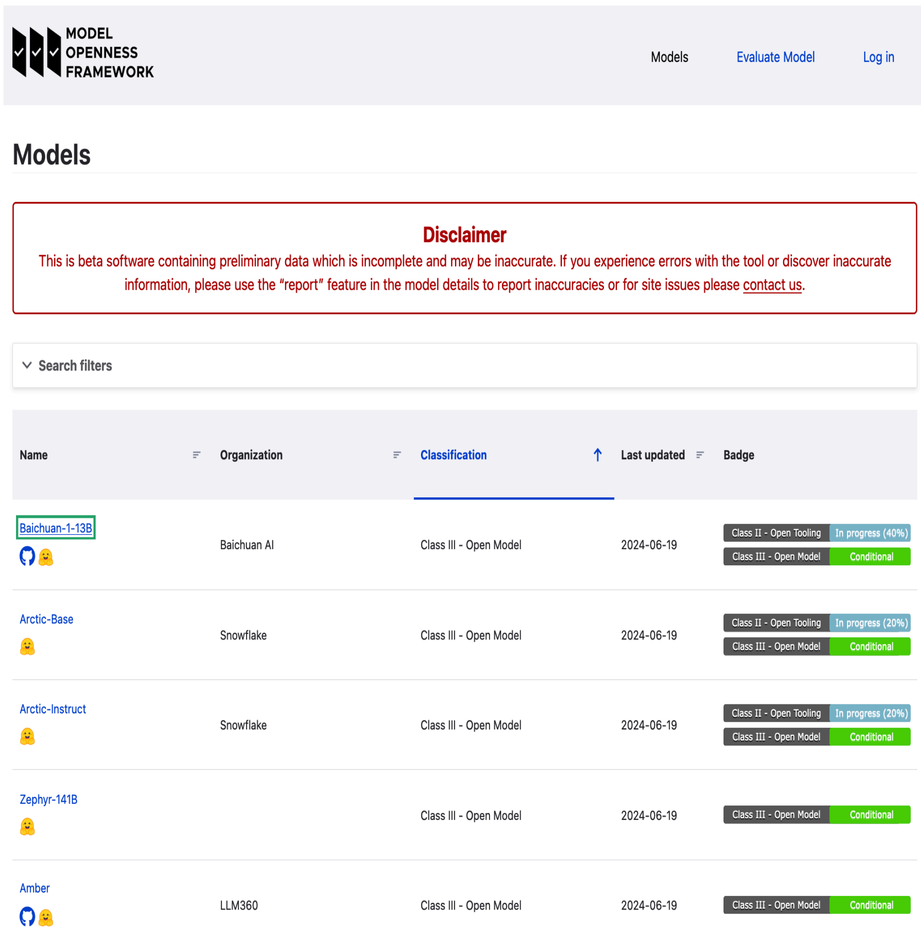

Figure 4: View models in the catalog of the Model Openness Tool.

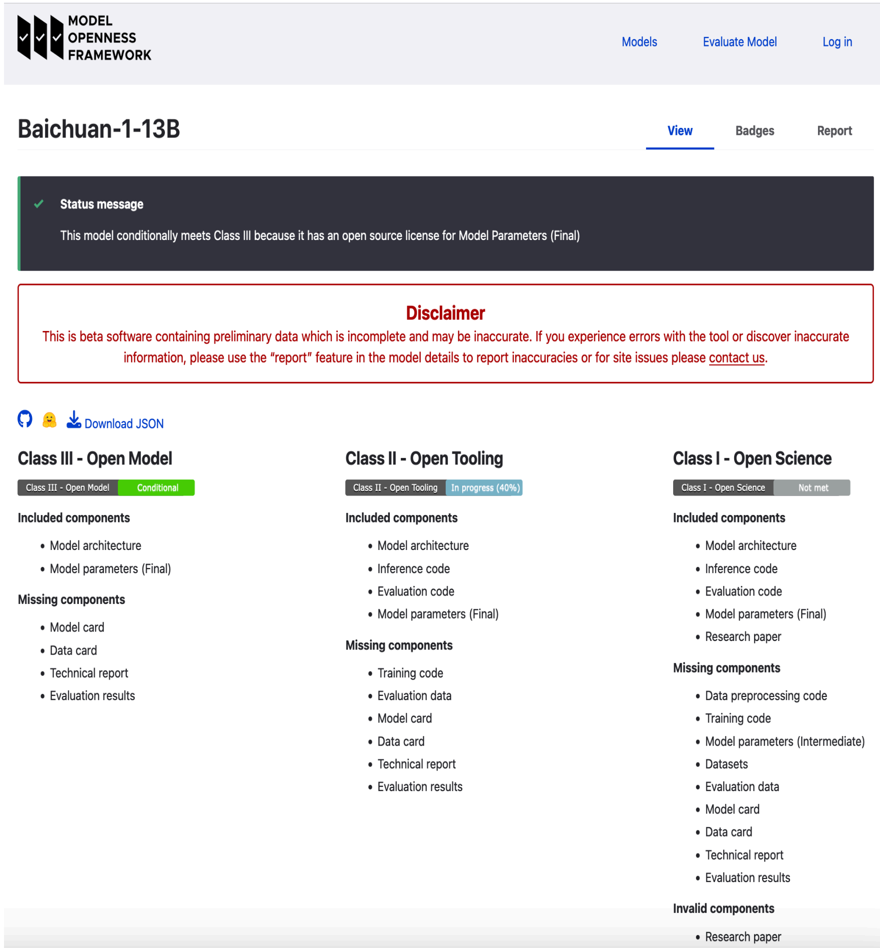

Figure 5: View a model's classification with the Model Openness Tool.

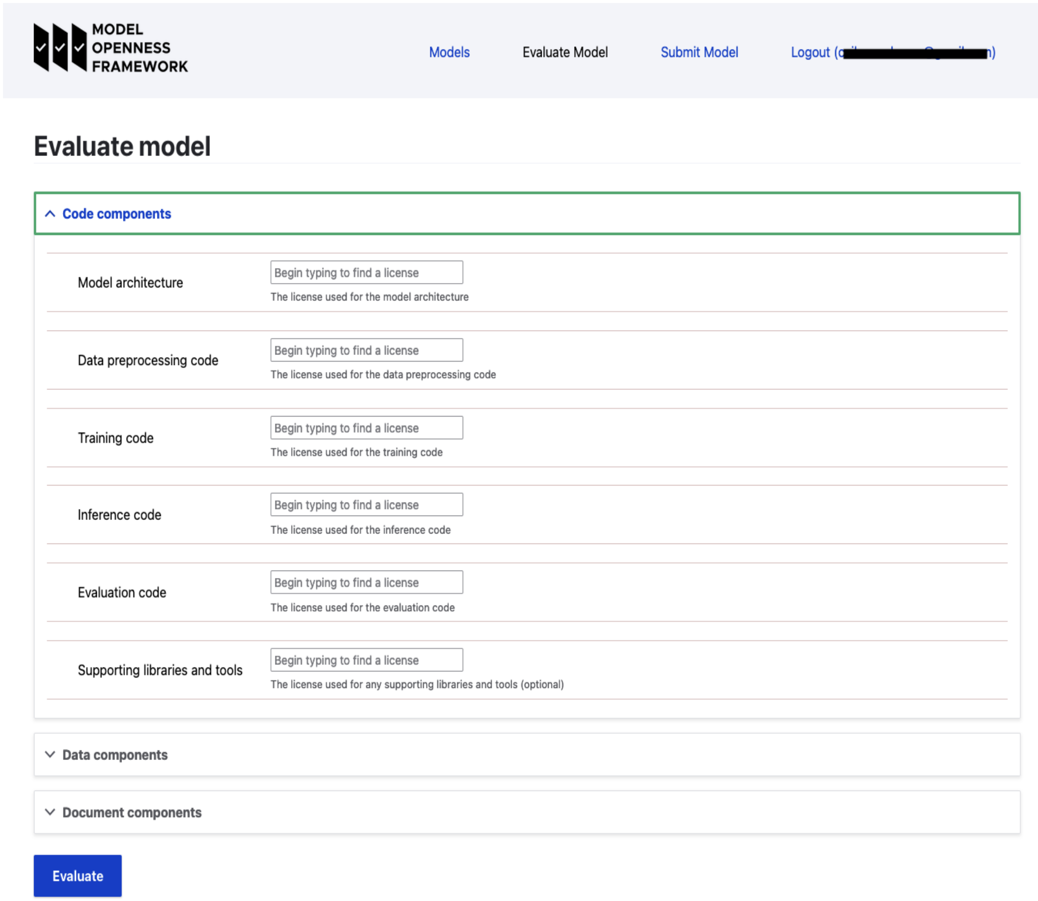

Figure 6: Evaluate models with the Model Openness Tool.

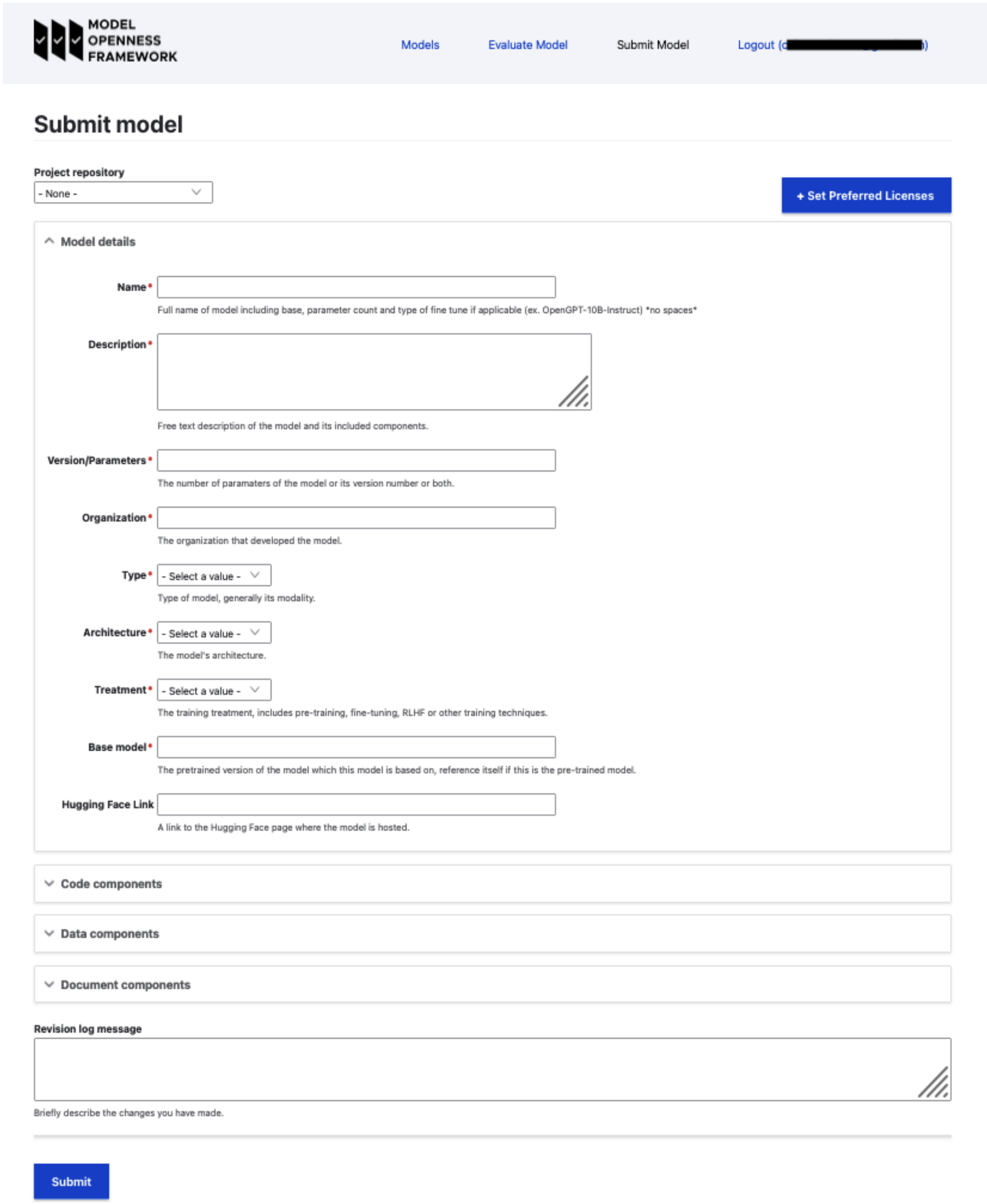

Figure 7: Submit models to the model catalog with the Model Openness Tool.

F Benefits of Model Openness Framework

The adoption of the MOF by the AI community brings many advantages, including but not limited to:

- **Clarity:** Clearly defines what components are included and under which license each is distributed, in order to understand the acceptable forms of use and whether a project is complete and truly open or not.
- **Openness:** By classifying models and their artifacts at increasing degrees of openness, the MOF will help push model producers towards creating the most complete and open models, helping to advance open science and both academic and commercial usage.
- **Reproducibility:** Comprehensive availability of data, code, and models enables others to independently reproduce results and identify sources of errors, bias or disparities. This strengthens scientific rigor.
- **Transparency & Explainability:** Opening model architectures, weights, training code, and documentation sheds light on how models work and behave. This builds appropriate trust and aids in inspecting for issues.
- **Data Provenance:** Origination and attribution can be determined when the data and its details are released. This can be helpful in tracing bias in models or identifying sources of PII leakage.
- **Accountability & Fairness:** Public data and models can be audited for unwanted biases and harms. Model producers can be notified of problems discovered by the community.
- **Continuous Improvement:** Model producers and consumers can build on open models instead of starting from scratch, accelerating innovation and progress in AI.
- **Collaboration:** Sharing open resources allows model producers and consumers across different fields and organizations to pool knowledge and capabilities.
- **Education & Learning:** Data, code, and models support teaching and learning about AI. Students, new researchers, and new developers can more easily enter the field.
- **Regulation:** Openness makes models more amenable to oversight and governance, unlocking policy options.

G Limitations and Criticisms

G.1 Known Limitations

We acknowledge several limitations and likely criticisms.

- The MOF is designed for deep learning artifacts, but does not transfer directly to every form of learning in AI. It is applicable to classical ML but does not translate entirely to all aspects of reinforcement learning.
- Model producers are expected to be honest about the availability of the components released with their models and the openness of licenses for each component as well as the completeness of both in their release.
- It requires convincing model producers who may be reluctant to share their work publicly without restrictions.
- Openness goals must be balanced with privacy, IP, institutional policies, and commercialization pressures.
- Classifying models ignores their actual functionality, and bias, safety, and other harms remain a concern. However, openness with models and data enables external audits of quality and completeness.
- Simplicity of classification may not capture all nuances. However, enhancement of the rubric may occur.
- It does not address the use of copyrighted materials in training data, an area currently being addressed through courts and legislation. The MOF requires data to be open using an open license; however, we encourage model producers to use authorized data in training models and respect copyrights [10].

G.2 Out of Scope

The MOF is not designed to solve all issues related to AI and openness, and its effective adoption will rely on the AI community to be transparent and honest in their reporting of the components of the models that they release and the licenses applied to each. The MOF does not intend to address any of the following as they are best addressed through alternative methods and means: AI safety (including bias, fairness, and trustworthiness), performance testing, red-teaming, security and privacy, components related to model serving, and model provenance.