

# An Empirical Analysis of Machine Learning Model and Dataset Documentation, Supply Chain, and Licensing Challenges on Hugging Face

TREVOR STALNAKER, William & Mary, USA

NATHAN WINTERSGILL, William & Mary, USA

OSCAR CHAPARRO, William & Mary, USA

LAURA A. HEYMANN, William & Mary, USA

MASSIMILIANO DI PENTA, University of Sannio, Italy

DANIEL M GERMAN, University of Victoria, Canada

DENYS POSHYVANYK, William & Mary, USA

The last decade has seen widespread adoption of Machine Learning (ML) components in software systems. This has occurred in nearly every domain, from natural language processing to computer vision. These ML components range from relatively simple neural networks to complex and resource-intensive large language models. However, despite this widespread adoption, little is known about the supply chain relationships that produce these models, which can have implications for compliance and security. In this work, we conducted an extensive analysis of 760,460 models and 175,000 datasets extracted from the popular model-sharing site Hugging Face. First, we evaluate the current state of documentation in the Hugging Face supply chain, report real-world examples of shortcomings, and offer actionable suggestions for improvement. Next, we analyze the underlying structure of the existing supply chain. Finally, we explore the current licensing landscape against what was reported in previous work and discuss the unique challenges posed in this domain. Our results motivate multiple research avenues, including the need for better license management for ML models/datasets, better support for model documentation, and automated inconsistency checking and validation. We make our research infrastructure and dataset available to facilitate future research.

## ACM Reference Format:

Trevor Stalnaker, Nathan Wintersgill, Oscar Chaparro, Laura A. Heymann, Massimiliano Di Penta, Daniel M German, and Denys Poshyvanyk. 2025. An Empirical Analysis of Machine Learning Model and Dataset Documentation, Supply Chain, and Licensing Challenges on Hugging Face. 1, 1 (December 2025), 48 pages. <https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

## 1 INTRODUCTION

The use of machine learning (ML) models in software applications has increased dramatically over the last decade, including in recommendation systems, computer vision, chatbots, image generation, and automated software engineering. As creating and training new models has become increasingly more expensive [92], developers have turned to fine-tuning pre-existing models. This approach

---

Authors' addresses: Trevor Stalnaker, William & Mary, Williamsburg, VA, USA, [twstalnaker@wm.edu](mailto:twstalnaker@wm.edu); Nathan Wintersgill, William & Mary, Williamsburg, VA, USA, [njwintersgill@wm.edu](mailto:njwintersgill@wm.edu); Oscar Chaparro, William & Mary, Williamsburg, VA, USA, [oscarch@wm.edu](mailto:oscarch@wm.edu); Laura A. Heymann, William & Mary, Williamsburg, VA, USA, [laheyrm@wm.edu](mailto:laheyrm@wm.edu); Massimiliano Di Penta, University of Sannio, , Italy, [dipenta@unisannio.it](mailto:dipenta@unisannio.it); Daniel M German, University of Victoria, Victoria, Canada, [dmg@uvic.ca](mailto:dmg@uvic.ca); Denys Poshyvanyk, William & Mary, Williamsburg, VA, USA, [denys@cs.wm.edu](mailto:denys@cs.wm.edu).

---

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM XXXX-XXXX/2025/12-ART

<https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

may save time and effort, but it also introduces the complexity of a new ML supply chain, complete with many novel challenges.

Supply chains for conventional software typically consist of software components, libraries, configuration files, and processes that contribute to the software [88, 109, 113]. However, for ML-intensive systems, the typical ML supply chain is more complex, comprising all elements that make an ML algorithm work [46, 99]. Creating a modern ML system involves the integration of ML models and traditional software components, taking into account dependencies between models themselves or between models and their training datasets. In addition, the training of ML models can rely on multiple datasets or dataset aggregates, and datasets, in turn, each have their own supply chain. Because proper license compliance requires developers to consider the permissions granted and requirements imposed by any individual component's license, the complexity of ML supply chains raises new and challenging issues, particularly given that many questions remain regarding the applicability of copyright law to generative AI [74].

Adaptation or reuse of existing models can also introduce degrees of complexity [31]. Models can be fine-tuned [26] or quantized from existing models [48]. (Fine-tuning allows a developer to use an existing model to create a model that specializes in certain tasks or to change the model's behavior, such as removing safety or ethical protections. Quantization compresses an existing model such that it can be run on smaller hardware with minimal performance loss.) Different ML models can be combined to form a single architecture that can, in turn, be reused or adapted [118]. Finally, the outputs from one model can be used to train another model, such as through synthetic data [79] or the use of distillation [56]. (Distillation is a technique where a smaller, simpler student model is trained to mimic the behavior of a larger, more complex teacher model.)

The ML supply chain, particularly those involving generative models [74], is not necessarily linear, progressing cleanly from one dependency or stage to the next. Instead, as outlined by Lee *et al.* [74], there can be branches and even cycles in the supply chain. For example, datasets can be used not only in the initial training of models but also for the fine-tuning of pre-existing base models, sometimes in multiple instances and by different developers. Additionally, some stages of the supply chain can backfeed into others, making relationships increasingly complex, if not recursive. For example, the outputs of a generative model can be added to pre-existing datasets and used for training future versions. The sheer volume of content from generative models makes this situation increasingly likely, if not inevitable.

Understanding these relationships is critical not only for license compliance, which requires a full understanding of the components used in a project and their associated licenses, but also for detecting, mitigating, and managing security threats involved with model reuse, such as weight poisoning attacks [72], data poisoning [50], and even malware hidden in model weights and assembled at runtime [110].

While previous work has considered the documentation [93], evolution [65], and environmental impact [20] of ML models, the complexity and challenges of the ML supply chain as a whole have not yet been fully explored. The supply chain relationships between models and dependent GitHub repositories have been explored by Pepe *et al.* [93] and Jiang *et al.* [66], but the relationships between ML models remain understudied.

To contribute to bridging this gap, we aim to investigate the emerging ML model supply chain on Hugging Face [57] (hereafter referred to as the "ML supply chain"), specifically the documentation practices of model owners, the structure and complexity of the supply chain itself, and the existence and prevalence of potential license compliance issues. Launched in 2016, Hugging Face is, as of September 2025, the largest repository of ML models and datasets [63, 65]. As with GitHub and traditional software, Hugging Face provides a central hub for model developers and data scientists to explore, share, and experiment with ML models. While other model forges—such as TensorFlow

Hub [51], PyTorch Hub [102], and Model Zoo [71]—exist, the Hugging Face platform has the widest reach, hosting more than 750K models and 175K datasets across many different use cases. Hugging Face therefore represents a robust opportunity to understand ML supply chains and analyze how developers interact with them.

We mine models and datasets hosted on Hugging Face, using the available information to construct a supply chain graph. We highlight different types of documentation inconsistencies that complicate the creation of this supply chain graph, discuss the underlying structure and features of the current ML supply chain on Hugging Face, and analyze the present licensing landscape. We note that proper documentation and documentation practices are foundational to mapping, and thus managing, the ML supply chain. Ultimately, we demonstrate that there is still much work to be done to facilitate better dependency/supply chain management on Hugging Face and in the ML ecosystem.

The main contributions of this paper are the following:

- A critical evaluation of the current state of documentation on Hugging Face.
- An analysis of the interactions between models and datasets in the ML supply chain on Hugging Face.
- An analysis of the licensing ecosystem on Hugging Face and potential compliance problems.
- An in-depth manual analysis of the top models and datasets on Hugging Face.
- Actionable suggestions for how model hubs can better facilitate supply chain management activities.
- A cleaned, reusable, and extensible dataset of models, datasets, and their relationships, and a tool suite capable of generating newer versions of said dataset [97].

## 2 BACKGROUND

In this section, we provide the background necessary to contextualize and understand our results. Specifically, we highlight the ML supply chain, software licensing, and documentation practices for ML components on Hugging Face.

### 2.1 The ML supply chain

The ML supply chain has become a complex structure with many steps and components, each of which can affect the final output of a given model. Lee *et al.* [74] identify eight events in the generative AI supply chain, but most apply to AI systems more broadly:

- (1) the creation of expressive works that will eventually be used to train a model;
- (2) the conversion of these expressive works into digital data;
- (3) the compilation of these data points into training datasets;
- (4) the creation of an ML model by selecting an architecture, training datasets, and a training algorithm;
- (5) the fine-tuning of existing “base” models;
- (6) the deployment of the trained/fine-tuned model as a service;
- (7) the use of the model to generate output; and, finally,
- (8) the application of additional alignment to further improve the model or meet user needs.

More generally, the ML supply chain comprises various components, including datasets, models, and traditional software. These components relate to each other in several key ways, including data examples being collected to create a dataset, multiple datasets being aggregated to create a composite dataset, datasets being used to train ML models, models being fine-tuned on pre-trained base models, and models being incorporated into software projects. These complex relationships distinguish the ML supply chain from other software supply chains: different steps in the process of creating and deploying ML models can feed back into each other in various, nonlinear ways, as

identified by Lee *et al.* [74], and different types of dependency relationships can relate to or build off of the original works in different ways, as identified by Duan *et al.* [32]. Understanding how work relates is important, as reusing any component through a supply chain can raise security risks or licensing implications [109].

Modern ML models are also becoming increasingly complex with respect to their architectures. For example, Mixture of Expert (MoE) models and ensemble models comprise multiple specialized models trained by dividing the problem space into homogeneous regions [80, 101]. Recent work has also proposed using smaller ML models to replace the hidden states within a Recurrent Neural Network (RNN) architecture to facilitate better recall and compression of context [98]. Both examples demonstrate that although these relationships have implications for effective supply chain management, understanding the relationships among models and model architectures will become increasingly difficult and may not yet be fully captured by systems such as the Hugging Face lineage graph utilized in this work.

Hugging Face, an online platform that hosts ML components, including models and datasets, has become a central hub for those who use and create ML components, showing exponential growth in recent years [22]. Users can download components provided by others and upload their own components. It has become the leading registry for pre-trained models online [67], presenting new opportunities for reusing ML components [31]. By studying the Hugging Face environment, we aim to uncover how multiple ML components can relate to one another and consider the implications of this critical supply chain. Specifically, we focus on the relationships between datasets and other datasets, between models and datasets, and between models and other models (events three, four, and five as outlined by Lee *et al.* [74]).

## 2.2 Software Licensing

In the same way as other creative works, software is protected by copyright law in the United States [5]. A copyright holder has several rights under U.S. copyright law, including the right to reproduce the work, create derivative works, distribute the work, and authorize others to engage in these activities [6]. Although the copyright owner can transfer some or all of the rights in a work, an owner who wishes to retain copyright while authorizing use by another party in a way that would otherwise be infringing typically does so by means of a license, which allows the copyright owner to impose conditions on the use [81]. In practice, developers often select pre-existing licenses drafted by other organizations and entities rather than drafting new licenses [8, 114]. This means that a particular license may or may not be a good fit for the subject of the license, the intended uses, or the copyright status of the underlying work.

The licensing landscape for ML models, particularly in Hugging Face, is very diverse, including open-source licenses (such as GPL and MIT) and Creative Commons licenses, as well as new ML-specific licenses such as OPEN-RAIL and LLAMA. These licenses broadly fall into two categories: permissive and restrictive (sometimes called copyleft). Permissive licenses (such as MIT [4], APACHE-2.0 [2], and CC-BY-4.0 [23]) typically impose few restrictions, often only requiring the user to provide notice files containing attribution or license information. On the other hand, copyleft licenses (such as GPL [3] and CC-BY-SA-4.0 [24]) typically require that derivative works be made available under the same or compatible licensing (leading some to characterize such licenses as “viral”).

ML-specific licenses follow a similar pattern, but often also impose additional restrictions on model usage, typically involving ethical issues, limiting commercial use beyond a certain number of active users, and prohibitions on using model output to train competing models [7, 86]. These additional restrictions have led some to consider many ML-specific licenses as falling outside

the open-source definition [112]. In response, more new licenses—such as OpenMDW [45, 47]—have emerged, implementing the Model Openness Framework [111], which aims to align model development with open-source principles.

License compliance is a task with which developers often struggle [106]. These challenges are likely heightened in the ML supply chain, given the uncertainties and ambiguities involved in interpreting OSS and CC licenses in the new ML context [32, 54]. Developers frequently treat license compliance as a lower priority, partly due to limited enforcement [114]. However, non-compliance can lead to financial [52, 104], legal [9, 11, 19, 27], and reputational [28, 53] consequences for developers and organizations, particularly organizations of sufficient size and prominence to attract attention for non-compliance. More specifically, to the extent that a licensed work is protected by copyright law, use beyond the scope of the license may constitute copyright infringement, which can result in litigation by the licensor, demands for payment or other remedies, and/or undesirable publicity that affects the licensee’s standing in the community.

### 2.3 ML component documentation

Comprehensive component documentation is essential to understanding the ML supply chain and maintaining license compliance. This need has become increasingly critical as the complexity of the components and the processes involved in their creation and deployment continue to grow. However, ML components do not always disclose data sources, making it difficult to comply with or keep track of all licensing obligations associated with each component [55]. This can create conditions that lead to legal disputes over the use of copyrighted material in training ML models [108], including not only the use of copyrighted works for training without permission but also the use of the generations of one model to train another model, an activity that may not have been permissible under that model’s licensing terms [82].

Model cards [85] have become the standard method for sharing information about ML models hosted on Hugging Face [42], including intended uses, limitations, and datasets used in training [41]. Although tools are in development to streamline the process [43], model card creation on Hugging Face is primarily a manual process in which a user enters documentation about a model into a predefined markdown template [36]. This template is robust, including fields for model description, uses, bias, limitations, testing, and optional fields for citation, technical specifications, and environmental impact. However, the manual nature of documentation introduces a significant potential for human error, ambiguity, and incompleteness [64, 66]. In fact, the quality and adoption of model cards remain low [16], despite user studies and other efforts by Hugging Face [34] to improve the situation, and despite researchers proposing tools such as DocML [16] to create and evolve model cards.

## 3 RELATED WORK

Hugging Face has been studied in recent years to investigate the maintenance and evolution of ML models [22], security risks in the ML supply chain [65], the carbon footprint of ML models [21], and the documentation of ML datasets and models [77, 91, 93].

### 3.1 The ML supply chain

By analyzing discussion forums and a mining study of Hugging Face, Taraghi *et al.* [100] identified various challenges and benefits associated with model reuse. However, they did not explicitly examine the dependency relationships that result from reuse. Other studies, such as those by Pepe *et al.* [93] and Jiang *et al.* [66], have primarily focused on the relationship between models and downstream GitHub repositories that utilize those models. In contrast, our work instead focuses on the relationships among pre-trained/base models, dependent models, and datasets. While Jones *et al.*

do consider model-to-model relationships, they stop short of constructing a comprehensive supply chain graph [67]. We extend this line of research by constructing such a graph and conducting a robust statistical analysis of its structure. We also conduct a detailed manual review of the most popular models (Section 8) and datasets hosted on Hugging Face (Section 9).

### 3.2 Software Licensing

Much previous work has investigated software licensing for traditional software, including software licensing issues [69, 106, 115] and practitioners' understanding of software licensing [8, 114]. More recently, some work has investigated the licensing issues that arise when ML components are used by open-source projects. Works by Pepe *et al.* [93] and Jiang *et al.* [66] explore model licensing on Hugging Face, primarily reporting the presence of licensing information as well as the usage frequencies for those licenses. In the case of Jiang *et al.* [66], this was done as a short demonstration illustrating the utility of their PEATMOSS dataset. Both works also consider the interplay of model licenses with the licenses of dependent GitHub repositories. Neither work, however, evaluates license incompatibilities between models or between models and datasets, as we do in this work. We extend these prior works by collecting and analyzing more recent data, which reflects the current state of model/dataset licensing.

The work of Duan *et al.* [32] proposes a tool (ModelGo) and framework for identifying potential license compliance issues involving ML components but provides only artificial examples and offers no empirical analysis of the prevalence of the potential problems in practice. Instead, we consider real-world examples of potential license conflicts and offer a rich, empirical analysis, including a report of license usage statistics, an examination of the frequencies of license differences between parent and children models, and thorough case-studies on a sample of the top models and datasets from Hugging Face. We further expand on prior work by considering various license classes, multi-licensing, and failing to comply with license-imposed naming requirements.

### 3.3 Model and dataset documentation

Previous research has identified several documentation shortcomings on Hugging Face. Works by Oreamuno *et al.* and Jiang *et al.* highlight that the majority of models and datasets hosted on Hugging Face lack documentation and that less-popular components are more likely to have incomplete documentation [63, 91]. Additional studies have explored the fields most likely to be missing from the dataset documentation [119], the presence and quality of the license information [77, 93], and the documentation of known model biases [93]. In addition, some studies have considered other deficiencies, such as poorly chosen or erroneous model names (*e.g.*, nonadherence to conventional naming schemes that lead to confusion) [60, 61].

We build on these previous analyses by identifying *specific* and *previously unreported* documentation failures, such as component reference ambiguities, encountered in our analysis of the ML supply chain on Hugging Face (Section 5). Our detailed analysis and enumeration of specific, concrete examples of incomplete and erroneous documentation go beyond previous work. These real-world observations can help model owners avoid common documentation pitfalls, empower model hubs to implement automated checks for common errors, and make researchers aware of potential noise in data mined from Hugging Face.

### 3.4 Summary

In conclusion, while our study builds on the basis of prior work [20, 60, 62–64, 91, 93, 100, 119], our investigation focuses on the structure and characteristics of the ML supply chain itself and the challenges encountered in managing it. More specifically, while previous work examined deficiencies in documentation and licensing at the level of individual models, we investigate the ML

supply chain at a higher level by examining how it is structured and how models relate. Additionally, where previous work has examined the relationships between projects and models, we examine the relationships among multiple models. Thus, our work is beneficial not only to software developers, as is the case for prior work, but also to data scientists, ML engineers, and anyone who wants to reuse models with complex dependencies. By providing an ecosystem-level examination of the ML supply chain, we provide these groups with the information necessary to understand how models relate to one another and evolve, licensing concerns regarding ML components on Hugging Face, and potential challenges one can face regarding model documentation.

## 4 STUDY METHODOLOGY

The *goal* of this study was to evaluate the current state of documentation, structure, and licensing of the ML supply chain. The study *context* consisted of 760,460 models and 175,000 datasets mined from Hugging Face. We aim to address the following research questions (RQs):

**RQ<sub>0</sub>:** *What documentation deficiencies exist in Hugging Face model and data cards that potentially complicate mapping the ML supply chain?* To understand the ML supply chain on Hugging Face, we needed to collect and analyze Hugging Face metadata. Using these data, we identified several issues and challenges related to documentation that introduced hurdles to obtaining a comprehensive understanding of the ecosystem. We examine these challenges in detail because with poorly and inconsistently described models and datasets, developers may struggle to leverage the ML supply chain effectively. This preliminary research question discusses examples of noisiness issues occurring in the Hugging Face model and the data set documentation.

**RQ<sub>1</sub>:** *What is the structure of the ML supply chain?* The ML supply chain, like any other supply chain, is characterized by dependencies across components. This research question looks at the complexities of such a dependency graph.

**RQ<sub>2</sub>:** *What is the current licensing landscape for models and datasets and what potential compliance challenges does it pose?* ML models are software, so software licenses govern their use and redistribution. As such, these models present new challenges in the area of license compliance. To understand these challenges, we examine the state of licensing in the ML supply chain on Hugging Face by 1) analyzing common licenses and license categories, 2) exploring licensing decisions made in the supply chain, and 3) investigating potential compliance challenges, such as incompatibilities between parent/child model licenses, ambiguities introduced by undeclared, unknown, or custom licenses, and uncertainty surrounding how license terms apply in context.

In the following subsections, we describe the study methodology, which is depicted in Fig. 1. We start by mining model and dataset lists. Next, for each model/dataset, we extract information necessary to construct the ML supply chain graph, recording and analyzing documentation deficiencies encountered during the process (*RQ<sub>0</sub>*). Then we systematically analyze and evaluate the structure of the resulting supply chain graph (*RQ<sub>1</sub>*). The same graph, along with the licensing information, is then used to address *RQ<sub>2</sub>*.

### 4.1 Data collection

To answer the research questions, we used the Hugging Face API to extract a list of all models publicly hosted on the platform. An initial list of 500,000 models was pulled on July 9, 2024, and a full list of all available models was obtained on July 11, 2024. Using the same APIs, we mined data (model cards and metadata) for each model on this list. Figure 2 shows a simplified representation of the relevant model metadata that we acquired from the Hugging Face API. For models that were unavailable through the API (as described in Section 5.1), we employed a web-scraping technique relying on a combination of Python’s `request` module and the `BeautifulSoup` HTML parser. Mining

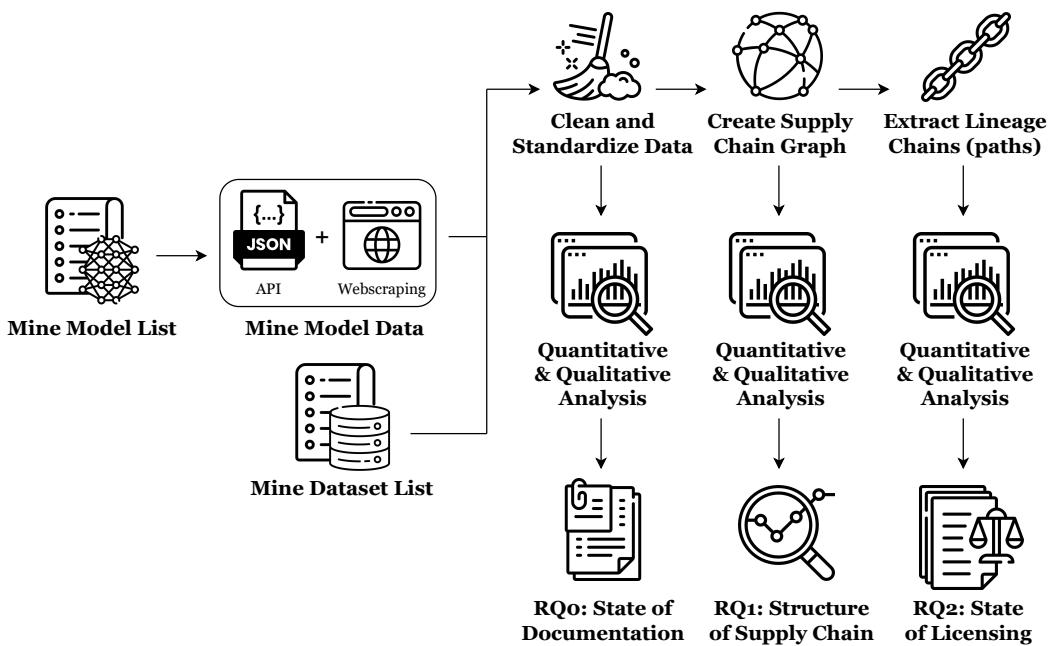


Fig. 1. Study methodology

took place between July 9 and July 12, 2024, yielding 760,460 mined models. We also used the Hugging Face API to gather the list of all datasets publicly hosted on the platform and available through the API as of July 9, 2024. For each model in our dataset, information on base models and datasets was obtained, when available, from the CARDATA attribute of the model metadata as well as from the model tags. We make all the raw data acquired from mining available in our replication package [97].

Although there are a handful of research datasets for Hugging Face models and datasets that already exist [64, 66], we opted to create our own dataset for this work for three reasons:

- (1) We aimed to analyze the most current data, which includes new models and model families (e.g., new models recently added to the Llama family of models) and changes in licensing frequencies (see Section 7). The presence of these new models is significant given that many of these were the first open-weight models capable of competing with the, until then, assumed more powerful and untouchable proprietary options (e.g., OpenAI’s Chat-GPT, Google’s Gemini, and Anthropic’s Claude). The practical utility of these new models represented a paradigm shift for fine-tuning, reusing, and remixing existing open-weight models. Their inclusion is essential to understand the present state of the ML supply chain on Hugging Face.
- (2) We aimed to collect and analyze a more comprehensive and complete dataset; as shown in Table 1, our dataset is significantly larger than most existing datasets and comprises both datasets and models.
- (3) Unlike Jiang *et al.* [66] and Pepe *et al.* [93], we were not focused on model-to-GitHub repository dependencies; accordingly, we aimed to design a dataset tailored for mapping model-to-model and dataset-to-model dependencies.

```

_id: "<internal Hugging Face ID>"
id: "<uploader name>/<model name>"
tags:
  0: "base_model:<name of base model>"
  1: "dataset:<name of dataset>"
  2: "license:<license identifier>"
downloads: <download count>
likes: <like count>
cardData:
  license: <license identifier>
  base_model:
    0: "<name of first base model>"
    1: "<name of second base model>"
datasets:
  0: "<name of first dataset>"
  1: "<name of second dataset>"
```

Fig. 2. Simplified representation of relevant metadata fields

Table 1. Comparison of study goals and Hugging Face (HF) datasets used in prior work

Work	Models	Datasets	Study Goal(s)
[119]	-	24,065	Explore the state of dataset documentation on HF
[60]	500	-	Investigate PTM* naming practices/anomalies on HF
[64]	15,913	-	Create a dataset for future research
[91]	55,280	6,758	Understand the state of model documentation practices
[63]	63,182	-	Understand model reuse, developer workflow, and challenges
[93]	159,132	-	Study extent to which model maintainers document PTMs
[100]	239,422	-	Understand challenges faced during PTM reuse
[62] / [66]	281,638	-	Create a dataset examining relationship between PTMs and OSS
[20] / [22]	380,000	-	Provide guide for researchers/explore how models are maintained
<b>This work</b>	<b>760,460</b>	<b>175,000</b>	Explore structure of ML supply chain and licensing challenges

\*Pre-Trained Model

## 4.2 Data normalization and cleaning

Once the data for models and datasets were downloaded, the data needed to be cleaned, standardized, and stored in a more usable form. This process involved standardizing the names of declared base models and datasets and mapping to unique identifiers where possible. For example, XLM-ROBERTA-BASE is a shorthand for FACEBOOKAI/XLM-ROBERTA-BASE, which has the unique internal ID 621FFDC036468D709F174364. We relied on Hugging Face's internal mechanisms to resolve short names where possible. For example, attempting to load the model page for XLM-ROBERTA-BASE on Hugging Face redirects to the page for FACEBOOKAI/XLM-ROBERTA-BASE. Additionally, we resolved inconsistencies in field inputs. For example, [], "", and NONE were all being used to denote no declared licenses. We standardized these variations to the empty list to make later computations simpler. We further elaborate on documentation challenges that required special attention and resolution in our discussion of RQ<sub>0</sub>.

Table 2. License information location for sample with no tags after manual review

Location	Count	Percent
Link to GitHub	9	56.3%
License referenced in README	2	12.5%
License text embedded in README	2	12.5%
LICENSE File	2	12.5%
License in tags (added after mining)	1	6.3%
Total	16	100.0%

#### 4.3 Extracting licensing information

We extracted licensing information from the tags present in model metadata. While licenses can also be declared in the CARDATA attribute of the model metadata, we found through an analysis of our dataset that the tags were more complete in every instance than the CARDATA attribute. In nearly all instances where licensing information was present, it was found in both locations (99.9%). However, for 0.1% of models, the licensing information could be found exclusively in the tags, and there were no instances that relied only on the CARDATA attribute. We also looked for discrepancies in the cases where both locations were utilized. We identified only 134 discrepancies, and in all cases, an additional license was declared in the tags that was not declared in the CARDATA. As there were no irreconcilable differences (*i.e.*, the tags declared an MIT license while the CARDATA attribute declared a GPL license), we concluded that the model tags are the most complete machine-readable source for licensing information.

Although licensing information can also be found in sections of a model card that are more difficult to programmatically extract from, we sought to further confirm that reliance on tags to identify the declared license used by the model was justified. Accordingly, one author manually reviewed the model cards for the top 100 models (sorted by number of downloads) marked as having no license in the tags. We anticipated that these 100 models were more likely to have complete and reliable documentation because of their popularity as compared with other models in our dataset, consistent with the findings of Jones *et al.* [67]. From this sample of 100 models, we identified only 16 models that had declared licensing information elsewhere. Of those, one had licensing information added to its tags after our initial mining, bringing the total to 15 models. For nine of these models, the license could only be found transitively by following a link to the model’s associated GitHub page. Table 2 provides a more detailed breakdown of where the licensing information was located for each example.

Detecting license declarations present in other sources (*e.g.*, plain text or links to external sites) would involve more complicated mining techniques, such as those that could traverse external links, and would require NLP-based approaches that extract licensing information from the text. These NLP approaches would have to not only recognize license names, which may not be in a standardized form, but also pattern-match embedded license text. An LLM tool, like the one employed by Jiang *et al.* [66], could potentially be used to extract licensing information, but at a high cost (between \$0.01 and \$0.03 per model—or between \$7,000 and \$22,800 for our dataset). Relatedly, relying on a cheaper but smaller, locally hosted LLM would risk introducing hallucinated license information and so still require laborious manual human oversight. To ensure that our analysis captured the most accurate information possible and to better understand the formal documentation structure of Hugging Face, rather than employ such an approach, we focus on explicitly declared licenses. Given the likelihood that licenses are declared in tags, the infeasibility of manual annotation for all examples, and the inherent difficulties of automatically obtaining license information from anywhere in a model card, our reliance on tags for license detection

appears to be an acceptable approach. Additionally, existing supply chain analysis tools rely heavily on machine-readable metadata, so we further justify that this is the appropriate locus of analysis given that the ultimate audience for such tools is the end user.

#### 4.4 Data analysis

To address **RQ<sub>0</sub>**, we quantitatively and qualitatively analyzed and discussed the different types of documentation issues that we observed in our attempt to understand the ML supply chain on Hugging Face.

Once the dataset had been cleaned, to address **RQ<sub>1</sub>**, we used the networkx Python library [87] to create a directed graph representing the observable ML supply chain. Due to the inherent issues and incompleteness of model documentation, this graph, by its nature, only represents a subset of the total ML supply chain on Hugging Face. Each node in the graph represents a distinct model, and an edge represents a dependency relationship. For example, if model *A* listed model *B* as a base model, there would be an edge from *B* to *A*. We would describe model *B* as the parent of model *A* and model *A* as the child of model *B*. That is, model *A* is derived from model *B*.

For the sake of analysis, we flatten the individual model supply chains by extracting all longest paths (*i.e.*, lineage chains) from the graph that go from source nodes to sinks. Each of these paths goes from a root base model to a model at the end of the ML supply chain (*i.e.*, a leaf node with no dependents or outgoing connections—no other models are derived from this model).

Using the paths resulting from **RQ<sub>1</sub>**, we addressed **RQ<sub>2</sub>** by analyzing the state of licensing in the ML supply chain, developing scripts and heuristics to look for parent/child license differences and potential license compliance problems related to model/dataset dependencies. We define a *parent/child license difference* to refer to the difference between the license(s) of a parent and a child model/dataset. Assume, for example, that model *B* is the child of model *A*. If model *B* was licensed under MIT and model *A* under APACHE-2.0, we would say there is a parent/child license difference between *A* and *B*. We avoid stronger terms such as “inconsistencies” and “incompatibilities,” since some of these differences may in fact be compatible with licensing terms. This is consistent with prior work [32] that found that the application of licensing terms, particularly for existing licenses, in the novel ML context can introduce uncertainties and ambiguities in terms of legal interpretation. Two authors reviewed all distinct licenses in our dataset and grouped them into six categories, which can provide the basis for future research into license compatibility and licensing trends in the ML supply chain.

All scripts used for mining, data clean-up, and analysis can be found in our replication package [97] and can be used to foster further research in this area.

### 5 RQ<sub>0</sub>: DOCUMENTATION CHALLENGES ENCOUNTERED ON HUGGING FACE

In this section, we outline and discuss documentation issues that were observed on the Hugging Face platform while mapping connections in the observed ML supply chain. Specifically, this preliminary RQ outlines the observed shortcomings and challenges to 1) elaborate potential pitfalls future researchers may face in mining Hugging Face and 2) to more thoroughly report documentation issues extant on Hugging Face that have been left unexplored by prior work.

#### 5.1 Inability to Access Metadata

Not all model metadata was accessible through the Hugging Face API. There were 7,258 model cards (0.95%) that had to be scraped using a combination of Python’s requests [94] and BeautifulSoup [14] libraries. The most frequent challenge, found in 99.4% of these cases, resulted from gating based on the acceptance of terms and conditions (see Figure 3). This was particularly the case for

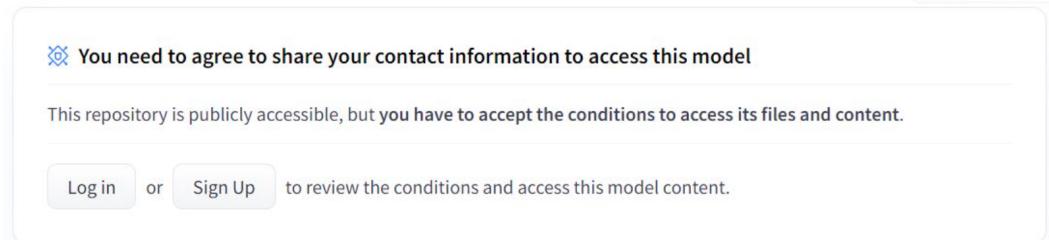


Fig. 3. Example of model gated behind Terms of Service requirement

models owned by or relying on models created by larger entities such as Google or Meta. Twenty-two models and their metadata were gated behind age restrictions imposed by the model owner\*, requiring a Hugging Face account to access. Accessing the full repository, and thus metadata, required logging in with a Hugging Face account and then, in most cases, providing an email address. These hurdles to accessing the metadata for a model could potentially complicate the creation of fully automated analyzers and ML/AIBOM (Machine Learning/Artificial Intelligence Bill of Materials) [12, 13, 96, 116] generators for an ML supply chain, which we believe should not rely on scraping potentially changing HTML content or on Selenium-based web-drivers for form completion and navigation.

## 5.2 Incomplete Metadata

We saw, consistent with prior work [63, 64, 66, 91], that the documentation provided by model owners on Hugging Face is often incomplete. This is a trend not just for smaller, infrequently used projects but also for large projects with tens of thousands of downloads. For example, OpenAI's CLIP-VIT-LARGE-PATCH14-336 model, with 5,827,027 downloads, has a model card that is almost entirely incomplete, save for basic hyper-parameter and framework version information [89]. Only 37.8% of models and 27.6% of datasets declared any kind of licensing information in a standardized, machine-readable way. We discuss the prevalence of declared components further in Section 6.

Some models may provide information on training data and architecture by linking to scientific papers that include these details, but such a practice introduces an additional hurdle in easily and programmatically obtaining that data, contributing to the ML reproducibility crisis [59]. Ultimately, the reason why many models do not even mention the datasets used would require further investigation, including interviewing/surveying developers, which is out of the scope of this work.

During the manual analysis of 100 model cards described in Section 4.3, we observed two examples where the Hugging Face team wrote documentation on behalf of model owners. The first instance was an image-to-text model uploaded by Microsoft with the disclaimer: "The team releasing TrOCR did not write a model card for this model so this model card has been written by the Hugging Face team" [83]. The second was a larger version of the same model, also owned by Microsoft [84]. In both cases, the models were described by a research paper and released on GitHub. Understanding the criteria that the Hugging Face team uses to determine when and how to intervene with model cards requires further investigation, but we do note that according to a blog written by members of the Hugging Face team, they created/updated model cards in some instances to inform design decisions surrounding a new model card template [33].

\*Throughout the paper, by "model owner", we refer to the entity that uploaded the model to Hugging Face. We make no claims as to whether that entity had the legal right to do so or if they are the original author/creator of the model.

### 5.3 The “Unknown” license

The Hugging Face platform’s documentation provides a list of recognized licenses, each with a unique, standardized short-hand identifier that can be utilized by users when creating their model/data cards [35]. This list includes common Open Source Software (OSS), Creative Commons (CC), and ML-specific licenses as well as an “Other” catch-all category that encompasses less common, custom, or modified licenses not included in Hugging Face’s predefined license list. An “Unknown” licensing option is also provided, but there is no guidance provided as to when this option should be selected. Future work will need to explore when and why model owners select this option, particularly given that acknowledging that the license is “Unknown” suggests, at best, a lack of due diligence or understanding on the part of the owner or, at worst, that the model/dataset raises copyright infringement issues. We observed 4,419 (1.5%) models and 2,194 (4.5%) datasets that used this “Unknown” license tag, which inherently raises compliance challenges for dependent models.

### 5.4 Naming problems

Hugging Face users primarily use human-readable names, not unique identifiers, when supplying the metadata information for their base models/datasets. Unique identifiers do exist for all models and datasets on Hugging Face, but we did not observe any instances where this information was being used by model/dataset owners when referring to models or datasets they depended on. Instead, these references typically, but not always, followed the “owner/model” naming convention. For example, a model owner would refer to a base model as “FacebookAI/xlm-roberta-base” and not by its unique Hugging Face ID: 621ffdc036468d709f174364. This is problematic because these model references are not automatically updated when name changes occur. A change to the model, dataset, or owner name results in a new human-readable identifier that no longer matches the previous references to the model/dataset. As a result, documentation can easily become outdated, unhelpful, and potentially confusing.

We observed 596 models that had their human-readable identifiers changed. Instances such as 222GATE/BLURDUS-7B-v0.1 → GATE369/BLURDUS-7B-v0.1 involve a change to the name of the model owner. Other instances, such as AADITYA/OPENBIOILLM-LLAMA3-70B → AADITYA/LLAMA3-OPENBIOILLM-70B, involve a change to the model name itself. In both cases, manual effort is required to track down and map the new names to the original references, which can make dependency management tasks increasingly difficult. Often the only way to determine the correct mapping is by relying on Hugging Face to redirect the page associated with a former name to the one associated with the most current name. This suggests that Hugging Face maintains some internal mapping, but it is unclear how long this mapping persists or what might happen in the event of a name collision.

Model names, without the additional owner information, are not guaranteed to be unique. Similar to how repository forks have the same name (but different owners) on GitHub, fine-tuned or forked models may also have the same name yet different owners on Hugging Face. For this reason, developers often specify models using the “owner/model” naming convention described previously. If this happens consistently, ambiguities can be avoided. However, we observed 16,477 cases where developers referred to base models only by the model name, excluding owner information. For example, a developer might use “roberta-base” to refer to “FacebookAI/roberta-base.” While it could be reasonable to assume that this shorthand refers to the popular model, this may not be the case: in our dataset, there were 39 other models also named “roberta-base,” all with different owners. Table 3 shows the top 10 most common model names in the Hugging Face ecosystem and their frequencies. If one of these models were referred to by name only, it would be nearly impossible to determine which model was being referenced. For example, more than 5,600 models share the

Table 3. Top 10 most common model names in the ecosystem

Model Name	Count
ppo-LunarLander-v2	5,660
q-FrozenLake-v1-4x4-noSlippery	4,200
test	3,900
ppo-Huggy	2,684
dqn-SpaceInvadersNoFrameskip-v4	2,405
distilbert-base-uncased-finetuned-emotion	2,348
my_awesome_model	1,727
q-Taxi-v3	1,535
ppo-SnowballTarget	1,513
bert-finetuned-ner	1,297

model name “ppo-LunarLander-v2.” While many of these are likely toy models or “Hello World” equivalents, others, such as DISTILBERT-BASE-UNCASED-FINETUNED-EMOTION, might suggest the need for widespread adoption of unique model identifiers.

### 5.5 Missing or nonsensical references

There were 34,159 instances where we were unable to map a declared dataset (*e.g.*, a dataset that a model reported relying on for training) to a dataset found publicly on Hugging Face. These consisted of 4,755 unique declarations. It is impossible to know with certainty, but such declarations likely consist of datasets that have been removed or made private, are from external sources, are actually dataset descriptors, contain typos, or, in some cases, are Hugging Face usernames.

We were also unable to map 2,501 base model declarations to known IDs, representing a possible 1,371 unique missing models in total. As with the datasets, these models may have been removed, made private, renamed in a way difficult to trace, or may reference a model outside the ecosystem. It is also possible that developer typos result in the inability to identify the models.

In other cases, some of the references were nonsensical. In our analysis of the graph of all collected components in the ML supply chain (described further in Section 6), we detected 684 cycles in total, including 675 (98.7%) trivial cycles in which a model declares itself as a base model, despite the fact that a model should not be able to be its own ancestor. All of this further motivates the need to use unique and standardized identifiers when referring to models and datasets, as well as a need for validating reference information.

### 5.6 Models as datasets

We observed 1,416 instances in which a model was listed as a dataset in the DATASETS field. In other words, a model reference was included in the metadata field specific to training datasets. Without more information, it is impossible to know whether this indicates a mistake on the part of the model owner or that the output of the declared model was somehow used for training.

### 5.7 Shortcomings of Hugging Face

While base models are declared by some model owners, in many cases, the precise relationship between a derived model and its base model is left ambiguous, at least in the available metadata. Hugging Face does provide a separate field for architecture relationships, but there is no standardized way to specify situations involving fine-tuning, quantization, or using outputs for training. This ambiguity can have significant consequences since, particularly in the licensing context, the nature of the relationship is important in determining how licensing terms should be applied. For example,

Table 4. Top 10 models by likes

Model	Pipeline	Likes	License
runwayml/stable-diffusion-v1-5	text-to-image	10,987	creativecommons-openrail-m
CompVis/stable-diffusion-v1-4	text-to-image	6,358	creativecommons-openrail-m
stabilityai/stable-diffusion-xl-base-1.0	text-to-image	5,502	openrail++
meta-llama/Meta-Llama-3-8B	text-generation	5,200	llama3
bigscience/bloom	text-generation	4,647	bigscience-bloom-rail-1.0
meta-llama/Llama-2-7b	text-generation	3,980	llama2
mistralai/Mixtral-8x7B-Instruct-v0.1	text-generation	3,960	apache-2.0
stabilityai/stable-diffusion-2-1	text-to-image	3,778	openrail++
WarriorMama777/OrangeMixs	text-to-image	3,714	creativecommons-openrail-m
meta-llama/Llama-2-7b-chat-hf	text-generation	3,670	llama2

Table 5. Top 10 models by downloads

Model	Pipeline	Downloads	License
MIT/ast-finetuned-audioset-10-10-0.4593	audio-classification	677,161,532	bsd-3-clause
sentence-transformers/all-MiniLM-L12-v2	sentence-similarity	83,805,061	apache-2.0
facebook/fasttext-language-identification	text-classification	53,517,573	cc-by-nc-4.0
google-bert/bert-base-uncased	fill-mask	52,724,375	apache-2.0
sentence-transformers/all-MiniLM-L6-v2	sentence-similarity	41,281,684	apache-2.0
openai/clip-vit-large-patch14	zero-shot-image-classification	40,114,370	-
distilbert/distilbert-base-uncased	fill-mask	24,195,717	apache-2.0
openai/whisper-small	automatic-speech-recognition	23,317,804	-
jonatasgrosman/wav2vec2-large-xlsr-53-english	automatic-speech-recognition	18,212,315	apache-2.0
openai/clip-vit-base-patch16	zero-shot-image-classification	16,999,162	-

it is permissible to use a model licensed under the LLAMA3 license for fine-tuning but impermissible to use its outputs to train a competing model [82]. Additional fields specifying the nature of relationships between models would be useful, but their inclusion would also introduce additional administrative burden for model owners.

While uncommon and not a direct shortcoming of Hugging Face, we observed examples where model owners would declare the same dataset or base model multiple times (*i.e.*, the same model/dataset had multiple entries in the respective metadata field). Specifically, 178 models declared a dataset at least twice, and 310 models declared a base model at least twice. One model, with otherwise robust documentation, declared the same base model 64 times. Again, while there are few instances of this duplication, they still serve as something of a canary in the coal mine, indicating a lack of validation on behalf of Hugging Face.

## 5.8 Summary

In summary, we identified numerous challenges while gathering data from Hugging Face, ranging from incomplete model information to the inability to access existing data. We highlight the prevalence of these issues to alert others seeking to mine Hugging Face and to identify specific areas for improvement, both from those documenting their models on Hugging Face and for any future efforts to improve Hugging Face’s system.

## 6 RQ<sub>1</sub>: ANALYSIS OF THE ML SUPPLY CHAIN

Despite the challenges outlined in RQ<sub>0</sub>, we were still able to construct an ecosystem-level supply chain graph mapping the relationships between models and datasets. While we acknowledge that our supply chain graph represents only a subset of the total components in the full Hugging Face

Table 6. Top 10 datasets by likes

Dataset	Declared task	Size	Likes	License
fka/awesome-chatgpt-prompts	Question Answering	<1K	5,011	cc0-1.0
HuggingFaceFW/fineweb	Text Generation	10B-100B	1,539	odc-by
Open-Orca/OpenOrca	Text Classification	1M-10M	1,269	mit
OpenAssistant/oasst1	-	10K-100K	1,232	apache-2.0
gsdf/EasyNegative	-	<1K	1,128	other
Anthropic/hh-rlhf	-	100K-1M	1,092	mit
togethercomputer/RedPajama-Data-1T	Text Generation	1M-10M	1,025	-
Nerfgun3/bad_prompt	-	<1K	921	creativecommons-openrail-m
tiiuae/falcon-refinedweb	Text Generation	100M-1B	765	odc-by
allenai/dolma	Text Generation	>1T	752	odc-by

Table 7. Top 10 datasets by downloads

Dataset	Declared task	Size	Downloads	License
hails/mmlu_no_train	Question Answering	* <sup>*</sup>	16,146,384	mit
lighteval/mmlu	Question Answering	1M-10M	5,882,652	mit
argilla/databricks-dolly-15k-curated-en	-	10K-100K	4,671,897	-
ceval/ceval-exam	Text Classification	10K-100K	2,111,340	-
lavita/medical-qa-shared-task-v1-toy	-	<1K	1,792,638	-
lukaemon/bbh	-	1K-10K	911,736	-
haonan-li/cmmlu	Multiple Choice/Question Answering	10K-100K	633,608	cc-by-nc-4.0
chansung/requested-arxiv-ids-3	-	<1K	587,249	-
cais/mmlu	Question Answering	100K-1M	555,115	mit
allenai/ai2_arc	Question Answering	1K-10K	543,678	cc-by-sa-4.0

\* No size is provided since the dataset requires Python code execution

ecosystem, as noted previously, we believe that it is appropriate to rely on it to make observations about recurring trends and patterns.

To analyze the structure of the ML supply chain on Hugging Face, we began by addressing the issues identified in **RQ<sub>0</sub>** using different strategies depending on the issue. Where possible, we employed web-scraping techniques to pull metadata for models that were inaccessible through the Hugging Face API (Section 5.1). Models where even this analysis was impossible were excluded from the analysis. We standardized model/dataset names, preferring the unique Hugging Face ID (Section 5.4). Where naming ambiguities were present (multiple models with the same name, but different owners), we assumed the most popular model (Section 5.4). References that could not be mapped to existing models/datasets on Hugging Face were also excluded from analysis (Section 5.5). Models that were listed as datasets were excluded from the dataset analysis (Section 5.6).

## 6.1 Most popular models and datasets

We begin with an overview of the landscape by describing the most popular models and datasets. We measure popularity in terms of engagement via the metrics provided by Hugging Face: likes and recent downloads. These metrics also serve our goal of capturing the current state of the ecosystem by highlighting what models and datasets are currently being engaged with. Tables 4 and 5 show the ten most engaged with models in our dataset by these metrics. Organizations such as OpenAI and Meta occupy several top positions. The models span a variety of use cases, including image and text generation.

Similarly, Tables 6 and 7 display the most engaged with datasets. The most common *Task* field is “Question Answering” for the most downloaded datasets and “Text Generation” for the most liked datasets. However, a variety of *Data Modalities* and *Tasks* are represented across both groups. For example, *Data Modalities* included both text and images, while *Tasks* included other specifics such

as “Token Classification.” In fact, the most liked dataset by far, FKA/AWESOME-CHATGPT-PROMPT, is a repository of curated ChatGPT prompts rather than a training dataset for models, which suggests that it may be more prudent to examine download count to determine the most popular datasets. This is further supported by the download count’s correlation with the number of client projects on GitHub [93].

## 6.2 Most depended upon datasets and base models

Our ability to evaluate the ML supply chain relies on the dependency information provided by model owners. On Hugging Face, this information is found in the model metadata, where model owners declare base models and datasets. Although many models on Hugging Face were not created independently, we note that only 117,245 out of 760,460 models (15.4%) declare any base model and, of these, most (111,568 | 95.2%) declare only one base model. Overall, most models do not declare any base models: with respect to the number of base models declared, the average, as well as the minimum, first quartile, median, and third quartile, are all 0. The maximum is 53 unique base models.

Table 8 lists the ten models that were most frequently declared as a base model by the models in our dataset, *i.e.*, the models that were most frequently cited as dependencies by other models. They exhibit a varying number of likes and downloads and primarily come from large providers such as Meta, Google, and OpenAI. Notably, none of these models declare that they are derived from any base models. We observe some overlap with the ten most engaged with models by downloads, suggesting, not surprisingly, that popular models are also frequently used as base models.

In addition to base models, we investigate the datasets used for model training. We observe that 75,516 (9.9%) models declare at least one dataset. Of these, 64,343 (85.2%) declare a single dataset, 5,066 (6.7%) declare two datasets, and 1,637 (2.2%) declare three datasets. The number of datasets also exhibits long-tail behavior, with one model (TASKSOURCE/DEBERTA-BASE-LONG-NLI) declaring 287 distinct datasets. However, the vast majority of models declare no dataset, with 0 declared models as the average, minimum, and first three quartiles, while the maximum is 287 datasets. We observe that this low percentage of declared datasets is particularly notable since, by their nature, nearly all ML models on Hugging Face should have been trained on at least one dataset, even if that dataset is not publicly available.

Table 9 describes the top ten declared datasets. We observe no overlap with the ten most liked or downloaded datasets. This may be a consequence of the low declaration rate for datasets, a function of a dataset’s age, or the fact that popular “datasets” uploaded to Hugging Face may not be intended for model training, as observed above.

License declarations vary among the datasets. Half of the ten most declared datasets are made available under a license declared as “Other.” One example from Google appears under six distinct licenses, including the APACHE-2.0 license, several Creative Commons licenses, and an “Other” license.

Table 8. Top 10 declared base models

Base Model	Times Used	Likes	Downloads	Pipeline	License
distilbert/distilbert-base-uncased	5,240	463	24,195,717	fill-mask	apache-2.0
stabilityai/stable-diffusion-xl-base-1.0	4,776	5,502	3,558,713	text-to-image	openrail++
runwayml/stable-diffusion-v1-5	2,438	10,987	3,790,115	text-to-image	creativeml-openrail-m
openai-community/gpt2	2,405	2,076	6,888,015	text-generation	mit
unsloth/llama-3-8b-bnb-4bit	2,363	141	486,478	text-generation	llama2
FacebookAI/xlm-roberta-base	2,169	526	6,196,387	fill-mask	mit
mistralai/Mistral-7B-v0.1	2,099	3,290	706,043	text-generation	apache-2.0
google-bert/bert-base-uncased	1,576	1,665	52,724,375	fill-mask	apache-2.0
google-bert/bert-base-cased	1,472	235	5,191,167	fill-mask	apache-2.0
openai/whisper-small	1,469	180	23,317,804	speech-recognition	apache-2.0

Table 9. Top 10 declared datasets

Dataset	Times Used	Likes	Downloads	License
nyu-mll/glue	2,293	336	122,473	other
rajpurkar/squad	1,827	228	11,379	cc-by-sa-4.0
dair-ai/emotion	1,766	257	23,236	other
ilsvrc/imagenet-1k	1,470	345	17195	other
mozilla-foundation/common_voice_7_0	1,330	163	67,565	cc0-1.0
stanfordnlp/imdb	1,329	208	20,078	other
legacy-datasets/wikipedia	1,219	528	1,657	cc-by-sa-3.0, gfdl
google/xtreme	1,154	82	982	apache-2.0, cc-by-4.0, cc-by-2.0, cc-by-sa-4.0, other, cc-by-nc-4.0
mozilla-foundation/common_voice_11_0	1,130	163	67,565	cc0-1.0
eriktks/conll2003	880	115	6,171	other

While these are all listed in the dataset’s metadata, the dataset card’s field for licensing information simply says, “More Information Needed,” leaving it unclear as to how these licenses jointly apply to the dataset. This suggests that even very popular components can contain ambiguities that make the license compliance process more difficult.

### 6.3 Structure of the supply chain

As explained in Section 4.1, our mined dataset consists of over 760,000 models. Our final model supply chain graph, which features the reported dependency relationships between models, features 761,826 nodes (models) and 126,995 directed edges. We remove any edges related to trivial cycles discussed in Section 4.3.3 where a model reports itself as its own base. This leaves us with 126,320 directed edges. The number of models in the graph exceeds the number in the dataset because some models were declared as dependencies but were not observed on Hugging Face at the time of mining. These models might have been removed, made private, exist on another platform, or had a name change that we were unable to trace.

The constructed supply chain graph can provide valuable information to model owners and researchers. For example, we can determine which models in the Hugging Face ecosystem are the most relied upon, and thus for which models complete, accurate documentation, compliance checks, and security audits may be more critical.

**6.3.1 Analysis of Removed Cycles.** As noted above, we removed from the analysis 675 trivial cycles. Of these, 168 featured nodes that were connected to other models in the supply chain graph. There were 154 instances (92%) where a model reportedly depended on itself and was also reported as a base for a another model. There were 14 instances (8%) in which a model reportedly depended on itself and at least one other model. In most cases, these cycles likely represent typos introduced by the model owner, but they could potentially also be artifacts indicating iterative fine-tuning or some other additional training.

**6.3.2 Analysis of graph degree and density.** There were 635,966 models (83.5%) isolated from the rest of the graph with no incoming or outgoing edges, representing the vast majority. In other words, they have a degree of zero. The next largest portion of nodes, a cohort of 111,840 (14.7%), have degree one. (“Degree” here refers to the total count of incoming *and* outgoing edges.) Table 10 shows the breakdown for statistics up to degree five. There was one model, DISTILBERT/DISTILBERT-BASE-UNCASED, that had an astounding degree of 5,240. For this model, all of these represented outgoing edges, meaning that 5,240 models recorded depending on DISTILBERT/DISTILBERT-BASE-UNCASED. Taking all this together, the graph’s density is very low ( $2.18 \times 10^{-7}$ ), reinforcing that most models

do not record any model dependency relationships. We rely on NETWORKX’s DENSITY function to calculate and return the graph density, using Equation (1).

$$D = \frac{|E|}{|V|(|V| - 1)}, \text{ where } |V| \text{ is the number of nodes and } |E| \text{ is the number of edges.} \quad (1)$$

Table 10. Nodes of degree  $N$  in model graph

Degree*	Node Count	Percentage
0	635,966	83.5%
1	111,840	14.7%
2	6,643	0.9%
3	2,550	0.3%
4	1,392	0.2%
5	764	0.1%

\*Degree is the total count of incoming and outgoing edges

Table 11. Subgraph order frequencies

Order	Count
1	635,966
2	3,550
3	1,349
4	803
5	519

There are 13,985 model nodes that have at least one outgoing edge, meaning they are depended upon by at least one other model. Almost half of these (7,032) have at least two outgoing edges. 620 model nodes have an outgoing degree of twenty or more, and 150 have an outgoing degree of one hundred or more. There are 116,576 model nodes that have at least one incoming edge, meaning they record depending on at least one base model. Only 5,673 of these records are based on two or more base models. All of this demonstrates, unsurprisingly, that models are more likely to rely on other models than they are to be relied on themselves. We also note that this shows that the machine learning ecosystem is upheld by more than just a handful of well-known and readily recognizable models, such as the 150 models with one hundred or more direct dependents.

**6.3.3 Connected components (subgraphs).** There are 643,653 connected components (*i.e.*, disjoint subgraphs) within our supply chain graph. Nearly all of these, 635,966 (98.8%), are trivial subgraphs including just a single node. An additional 3,550 subgraphs are made up of only two nodes. Table 11 breaks down the most common order frequencies for the subgraphs. From this, it is clear that the connected components constructed based on available documentation tended to be small, but there were 107 subgraphs that contained one hundred or more nodes.

**6.3.4 An analysis of the largest subgraph.** The largest subgraph featured 28,879 nodes and 35,633 edges, giving it a density of  $8.54 \times 10^{-5}$ . The majority of nodes in this graph, 22,099 (76.5%), have a degree of one. This is followed by 3364 (11.6%) nodes with degree two and 1235 (4.3%) nodes with degree three. There were multiple hubs within the supply chain that served to connect the various nodes. For example, there were 81 nodes that had a degree greater than fifty. These represent popular models that have many models fine-tuned or based on them. See Table 8 for details on the top ten such models. Ultimately, we see that the sparse nature that is representative of the entire Hugging Face supply chain graph also applies to its largest connected component.

## 6.4 Model lineages

**6.4.1 Lengths of model supply chains.** We define a lineage chain as a path from some root base model (*i.e.*, one with no model dependencies) to a final sink node model (*i.e.*, one without any model dependents). We found and examine 53,151 lineage chains for models that declare at least one base model, including cases where multiple chains lead to the same model. Of these, the average chain

length to reach that model is 6.2 models. The most frequent chain length is three models (12,480 chains); the minimum is 2 models, the first quartile is 3, the median is 4, the third quartile is 9, and the maximum is 40.

**6.4.2 Licenses present in model supply chains.** There are typically relatively few parent/child license differences between models in these chains. We remind the reader than a parent/child license difference does not necessitate an incompatibility, but instead demonstrates that some licensing decisions (or lack thereof in the case of neglecting to license a derivative work) were made along the chain. There was an average of 1.9 distinct licenses in a given chain, including unknown or undeclared licenses. The minimum and first quartile are both just 1 license, both the median and third quartile are 2 licenses, and the maximum is 6 licenses. A chain most commonly contains 2 licenses (24,069 chains), and the highest number of licenses observed was 6, occurring in 17 different chains. A relatively small number of parent/child license differences simplifies the task of managing license compatibility. However, these differences within a lineage necessitate examination of license compatibility, as well as a possible need to find mitigations for items in model chains when required by the license of a base model.

**6.4.3 Model ownership.** We also observe that when choosing models to build on, developers using Hugging Face often use models from the community rather than using their own previous work as a base. The owners of nodes at the end of a chain appeared within that node's chain only 1.4 times on average, showing that base models are frequently sourced from the community and supporting the idea that a common approach to training a model involves building on previous work. Alternatively, it could be that model trainers do not provide base model information when building on their own work at least some of the time. Future work involving discussions with developers could shed light on the reasons behind this phenomenon.

In 82.9% of chains (43,507), the owner of the final model in the chain owned no other model in that chain. In the maximum case, the chain contained 39 models from the final model's owner. The minimum, first quartile, median, and third quartile are all a single occurrence of the final model's owner in a given chain: only the final model itself.

Based on our observations, a given owner publishes relatively few models to Hugging Face, owning, on average, 4 models in our dataset, though the data is skewed heavily toward relatively few highly-prolific owners. Of 190,136 distinct model owners in our dataset, 61.5% owned just one public model (117,016), while the most prolific account owned 4,610 models. The first quartile and median are both just 1 owned model, and the third quartile is 2 models. Notably, the large players mentioned above, including Meta and OpenAI, are not present in the top 10 most prolific model owners: those spots instead go to smaller stakeholders in the AI market. The top ten model and dataset owners on Hugging Face are shown in Table 12. There is one overlap between the two lists.

## 6.5 Summary

Our findings show that Hugging Face components rarely declare datasets, generally have short chains of related components, and when dependency relationships between components exist, licensing information usually stays the same from parent to child. This perhaps simplifies the task of managing dependencies and licensing, but largely by virtue of the fact that relevant supply chain information appears to be missing in many such cases.

## 7 RQ<sub>2</sub>: LICENSING OF MODELS AND DATASETS ON HUGGING FACE

Only 37.8% of models and 27.6% of datasets declared licensing information in a machine-readable way (see Section 4.3). In this section, we investigate this licensing information to better understand how datasets and models are licensed in the ML supply chain. We begin by organizing the licenses

Table 12. Top 10 model and dataset owners on Hugging Face

Models		Datasets	
Owner	Model Count	Owner	Dataset Count
mradermacher	4,610	open-llm-leaderboard-old	7,059
thebloke	3,863	cyberharem	3,384
huggingtweets	3,830	autoevaluate	3,232
tkcho	3,810	recag	2,345
jeevesh8	3,723	liuyanchen1015	1,558
prunaaai	3,411	adAPTEROCEAN	810
lonestriker	3,241	tyzhu	807
sail-rvc	3,182	joey234	705
larryaidraw	3,046	lots-of-loras	500
cyberharem	2,978	results-sd-v1-5-sd-v2-1-if-v1-0-karlo	462

that we observed into six categories, intended to be indicative of the origin or purpose of the licenses (whether the licenses were designed for open-source software, for ML components specifically, or for other purposes) and are not intended to convey any legal characteristics, particularly given the uncertainty involved in applying these licenses in the ML context. These categories are not aimed at resolving compliance issues but instead demonstrate the complex and diverse landscape of licenses in the ML supply chain. We define the following categories:

- OSS: open-source software licenses such as MIT, APACHE, and GPL
- CC: Creative Commons licenses
- ML: ML-specific licenses such as OPEN-RAIL and LLAMA
- Data: licenses that are intended to specifically protect data such as ODBL and C-UDA
- Other: the “Other” license category on Hugging Face
- Unknown: the “Unknown” license category on Hugging Face

## 7.1 Most common licenses

There were 71 unique declared licenses for models and 70 for datasets. Tables 13 and 14 provide an overview of the ten most frequently observed licenses.

We classify licenses by the categories defined above. A breakdown of the prevalence of these classes can be found in Table 15. Both models (62.5%) and datasets (58%) were most often licensed under approved OSS licenses. This is particularly interesting with respect to datasets, as we would expect to see primarily CC or Data licenses, as datasets are not “software” in a strict sense of the word. This may reflect that Hugging Face developers view models and datasets more similarly to programs than to data or that developers rely on OSS licenses because they are already familiar with their terms. However, as prior work indicates [32], the relationships between these preexisting licenses and these new types of components in the ML supply chain, as well as the legal and other implications of licensing such components with such licenses, are currently unknown. Further research is needed to understand the reasoning behind these licensing choices, as well as to understand the interactions between many of these licenses and ML components.

Both models and datasets are distributed with (i) OSS licenses specific to software (both restrictive and permissive), (ii) Creative Commons (CC) licenses, which are not software-specific yet are being used for software (e.g., Stack Overflow’s adoption of CC licenses), and ML-specific licenses. For models, our findings are consistent with those of Pepe *et al.* [93], showing that permissive OSS licenses such as APACHE-2.0 and MIT are popular in this space. This is not surprising, as such licenses are also among the most popular for open-source projects [105, 107] and allow

Table 13. Top 10 licenses for models.

Models			
License	Type	Count	Percent (of licensed)
apache-2.0	OSS	119,449	41.6%
mit	OSS	51,184	17.8%
openrail	ML	30,095	10.5%
creativeml-openrail-m	ML	20,396	7.1%
other	Other	16,447	5.7%
cc-by-nc-4.0	CC	8,318	2.9%
llama2	ML	6,158	2.1%
unknown	Unknown	4,419	1.5%
cc-by-4.0	CC	4,269	1.5%
llama3	ML	3,335	1.2%

Table 14. Top 10 licenses for datasets.

Datasets			
License	Type	Count	Percent (of licensed)
mit	OSS	13,501	27.9%
apache-2.0	OSS	12,372	25.6%
openrail	ML	5,501	11.4%
cc-by-4.0	CC	2,673	5.5%
unknown	Unknown	2,194	4.5%
other	Other	1,837	3.8%
cc-by-sa-4.0	CC	1,256	2.6%
cc-by-nc-4.0	CC	1,014	2.1%
cc-by-nc-sa-4.0	CC	946	2.0%
cc0-1.0	CC	799	1.7%

Table 15. Prevalence of license classes

Models			Datasets		
License Class	Count	Percent	License Class	Count	Percent
OSS	179,421	62.5%	OSS	28,047	58.0%
ML	66,540	23.2%	CC	9,151	18.9%
CC	19,856	6.9%	ML	6,481	13.4%
Other	16,447	5.7%	Unknown	2,194	4.5%
Unknown	4,419	1.5%	Other	1,837	3.8%
Data	326	0.1%	Data	559	1.2%
Combination	134	0.05%	Combination	87	0.2%
Total Licensed	287,143	100.0%	Total Licensed	48,356	100.0%

commercial/closed-source exploitation. We note that, in general, the top licenses adopted on Hugging Face are traditionally understood to be more permissive, even if their exact application in this novel context is not fully understood.

We also note the adoption (for models, but especially for datasets) of different variants of the CC license, including a permissive license (CC-BY), restrictive licenses (CC-BY-SA and CC-BY-NC-SA), and licenses limiting non-commercial use only (CC-BY-NC).

Besides OSS licenses, we see a large proportion of different kinds of ML-specific licenses. These include both licenses originating from open-source initiatives (e.g., OPENRAIL and CREATIVEML-OPENRAIL-M), and licenses originating from companies (e.g., Meta’s LLAMA2 and LLAMA3 licenses). Such licenses tend to have terms that differ from typical redistribution terms established by software licenses such as, for example, terms that require responsible model usage or prohibit adoption for harmful or unethical uses or uses that do not take models’ limitations into account. Likewise, the LLAMA2 and LLAMA3 licenses impose limitations related to commercial use, use in products or services having more than 700 million monthly active users, and use to train other models, unless such models are redistributed, under the same licenses, as derivative works of LLAMA models.

Lastly, we note the increased prevalence of Hugging Face’s “Other” license, which was ranked 7th in a study conducted by Pepe *et al.* [93], but has since moved to the 5th spot since. In fact, 3.8% of datasets and 5.7% of models are licensed with some “Other” license.

## 7.2 Exploration of the “Other” License

In our entire dataset, 5.7% of licensed models were labeled as being under some “Other” license. This makes the “Other” category the fifth most common licensing choice for models on Hugging Face, representing over sixteen thousand models. Non-standard, custom/vanity licenses can complicate compliance processes for organizations and individuals, given that the documents no longer represent a known quantity [114]. To assess the nature and potential impact of these licenses, we investigate these “Other” licenses in further detail.

**7.2.1 Manual analysis of models under the “Other” license.** We examine the licenses of the top 25 “Other”-licensed models, sorted by downloads, as well as those of 25 additional, randomly sampled “Other”-licensed models from the total population. Our assumption is that the popularity of models in the high-downloads group indicates a tendency toward high-quality documentation practices, allowing us to investigate a detailed and meaningful sample of projects licensed in this way while maintaining a reasonable sample size for manual analysis. We also select a random sample in an attempt to determine if the same licensing trends exist across the ecosystem, even for less popular models. One author manually and systematically reviewed the model pages of these 50 models to identify more specific licensing information. Cumulatively, these 50 models represent 13,945,199 downloads (0.8% of all model downloads). The results can be seen in Tables 16 and 17. (One of the models randomly sampled was no longer available on Hugging Face for analysis.)

**7.2.2 Top 25 “Other” licensed models.** For the top 25 models, additional licensing information was most often found in a LICENSE file that was associated with the repository, but four (16%) linked to an external page that included more licensing information. There were four cases (16%), including three Meta models, where a specific license could not be determined, but one could be inferred from a related GitHub repository, which often was not directly linked to the model card. We also observed a case, LYKON/DREAMSHAPER, where the license that was specified in the model card did not match the external reference also provided.

Determining the licenses for some models required looking beyond the information provided on Hugging Face itself. For example, the README for the model DIGIPLAY/ABSOLUTEREALITY\_v1.8.1 included a link to a model page on an external site (Civitai.com). On this page, the license is reported as CREATIVEML OPEN RAIL-M ADDENDUM. Originally, clicking the link associated with the license would take the user to a page for the ABSOLUTE REALITY LICENSE (a confusing license mismatch),

Table 16. “Other” license information for Top 25 “Other”-licensed models

Model Name	Resolved License	Find Location
facebook/opt-125m	OPT-175B LICENSE AGREEMENT	LICENSE.md
stabilityai/sdxl-turbo	sai-nc-community	LICENSE.md
coqui/XTTS-v2	coqui-public-model-license	LICENSE.txt
facebook/mask2former-swin-base-coco-panoptic	None Available*	None Available
briaai/RMBG-1.4	BRIA 2.0 on HuggingFace Model License Agreement	External Link (connected to tag)
facebook/mask2former-swin-large-cityscapes-semantic	None Available*	None Available
apple/DFN5B-CLIP-ViT-H-14-378	apple-sample-code-license	LICENSE
facebook/opt-1.3b	OPT-175B LICENSE AGREEMENT	LICENSE.md
facebook/mask2former-swin-large-ade-semantic	None Available*	None Available
Lykon/DreamShaper	CreativeML Open RAIL-M Addendum*	External Link (in model card)
Salesforce/instructblip-vicuna-7b	None Available*	None Available
SanctumAI/Meta-Llama-3-8B-Instruct-GGUF	llama3	LICENSE
cagliostrolab/animagine-xl-3.1	faipl-1.0-sd	External Link (connected to tag)
stabilityai/stable-video-diffusion-img2vid-xt	stable-video-diffusion-community	LICENSE.md
apple/mobilevit-small	apple-sample-code-license	In model card
jartine/gemma-2-27b-it-llamafile	Gemma License	In model card*
facebook/opt-350m	OPT-175B LICENSE AGREEMENT	LICENSE.md
Qwen/Qwen2-72B-Instruct-AWQ	tongyi-qianwen	LICENSE
Mozilla/Meta-Llama-3-70B-Instruct-llamafile	Llama3	In tags*
digiplay/AbsoluteReality_v1.8.1	Absolute Reality License*	External Link (in model card)
qwp4w3hyb/Qwen2-72B-Instruct-iMat-GGUF	tongyi-qianwen	In tags*
google/mobilenet_v1_0.75_192	None Available	None Available
Qwen/Qwen2-72B-Instruct	tongyi-qianwen	LICENSE
mradermacher/QuartetAnemoi-70B-t.0.0001-i1-GGUF	None Available	None Available
SanctumAI/Codestral-22B-v0.1-GGUF	mmp1	In tags*

\* Information was ultimately inconsistent, unavailable, or changed since initial mining.

Table 17. “Other” license information for random sample of “Other”-licensed models

Model Name	License	Location
Shalie/EliraPendoraPonyXL	faipl-1.0-sd	External Link (connected to tag)
sukhoi37/sp_weshi	None Available*	None Available
Bobross69/Usgiyam	None Available*	None Available
baratk/barat	Empty file	LICENSE file
stylalai/competition-math-phinetune-v1	None Available*	None Available
divyanshu074/Text_Beat	None Available*	None Available
LoneStriker/WhiteRabbitNeo-33B-v1-3.0bpw-h6-exl2	deepseek	External Link (connected to tag)
juhai/llama3-ift	REMOVED*	REMOVED*
KnutJaegersberg/Deacon-34B-qlora	yi-license	LICENSE file
second-state/Qwen1.5-1.8B-Chat-GGUF	tongyi-qianwen-research	LICENSE file
psychicfire/sample	None Available*	None Available
Workaholic/fake-pvc-style	None Available*	None Available
azi111/dolphin-2_2-yi-34b-465bpw-h8-exl2-cnen	Empty file	LICENSE file
dranger003/deepseek-coder-33b-instruct-iMat.GGUF	deepseek	External Link (connected to tag)
optimai/porne-0	None Available*	None Available
cortecs/Meta-Llama-3-70B-Instruct-GPTQ-8b	None Available*	None Available
TheBloke/MergeMonster-13B-20231124-GPTQ	Other / llama2	Model card
yzhuang/Meta-Llama-3-8B-Instruct_fictional_Chinese_v3	None Available*	None Available
fasterinnerloopr/stable-code-3b	None Available*	None Available
DevQuasar/Qwen2-72B-Instruct-GGUF	tongyi-qianwen	External Link (connected to tag)
titan087/Llama3-70B-ShiningValiant2-exl2-4b	llama3	External Link (connected to tag)
nulltella/phi-1_5-alcapa-51k-instruct	None Available*	None Available
Holarissun/mar13_gemma2b_aist_gsm8k_rand_alpha0.9995-subset7000	None Available*	None Available
raincandy-u/Quark-464M-v0.1.alpha	tongyi-qianwen-research	External Link (connected to tag)
TheBloke/Tess-34B-v1.4-GGUF	yi-34b	External Link (connected to tag)

\* Information was ultimately inconsistent, unavailable, or changed since initial mining.

but, in the time since, the link has been corrected to point to the full text of the CREATIVEML OPEN RAIL-M ADDENDUM.

We can also see there were three instances where a model reported an “Other” license, but the actual license specified was in Hugging Face’s recognized license list (either LLAMA3 or GEMMA). This could be the result of the model owner uploading their model before the license was officially recognized or a general unfamiliarity on the uploader’s part with the available license options. In either case, this particular error further underscores the manual and error-prone documentation processes on Hugging Face.

Somewhat surprisingly, even after systematic manual documentation searches (including web searches for associated GitHub repositories), there were two models in the top 25 “Other” licensed models (8%) where the specific license could not be identified. From a practical standpoint, these models might as well not be licensed at all, since users cannot comply with unavailable terms and conditions.

Interestingly, two of the models that had no licensing information included in their documentation, SALESFORCE/INSTRUCTBLIP-VICUNA-7B and GOOGLE/MOBILENET\_V1\_0.75\_192, actually had model cards produced by the Hugging Face team.

**7.2.3 Random sample of 25 Other-licensed models.** Next, we examined an additional random sample of 25 models that were reportedly under an “Other” license. This would allow us to determine if the licensing practices followed by the top 25 are representative of the ecosystem. As one might expect, we found that identifying the licenses for many of these models was difficult. Ten models (40%) had model cards that were void of all text. One additional model had some information in its model card but none that pertained to licensing. Two more models had LICENSE files in their repositories, but the files were empty. Another model, META-LLAMA-3-70B-INSTRUCT-GPTQ-8B, provided no further licensing information, but given its name, one might infer it was under the LLAMA3 license.

Some of the models, however, did include more specific licensing information. Six (24%) had links to external references included in their metadata. One model, while labeled as licensed under “Other,” was actually under the LLAMA3 license. Two models had LICENSE files that provided additional license information.

In the text of the model card for THEBLOKE/MERGE MONSTER-13B-20231124-GPTQ, we observed the model owner grappling with certain ambiguities and uncertainties surrounding the “Other” license: “The creator of the source model has listed its license as other, and this quantization has therefore used that same license. As this model is based on Llama 2, it is also subject to the Meta Llama 2 license terms, and the license files for that are additionally included. It should therefore be considered as being claimed to be licensed under both licenses. I contacted Hugging Face for clarification on dual licensing but they do not yet have an official position. Should this change, or should Meta provide any feedback on this situation, I will update this section accordingly. In the meantime, any questions regarding licensing, and in particular how these two licenses might interact, should be directed to the original model repository.” Here, we see that properly maintaining license compliance can be a difficult and confusing task, particularly when all relevant information is not made readily available. It is also important to note that since the original model being quantized was itself a derivative of a LLAMA2 model, this model uploader has restored the LLAMA2 license to remain compliant.

**7.2.4 Automated analysis of models under the Other license.** Having determined from the manual review of 50 models reportedly under the “Other” license how and where to find more specific licensing information, we next moved to an automated approach that could provide insights about the most common “Other” licenses in the ecosystem. We note that since this analysis required

Table 18. “Other” licenses reported in CARDATA attribute in Top 10,000 “Other”-licensed models

License Name	Count	Percentage
faipl-1.0-sd	2346	23%
flux-1-dev-non-commercial-license	1172	12%
llama3	269	3%
deepseek	206	2%
tongyi-qianwen	201	2%

Table 19. Parent/child license difference categories and their frequencies

Difference Type	Description	Example	Count	Percentage
No Overlap	Both the parent and child are licensed, but there is no overlap in their licensing	cc-by-nc-4.0 → apache-2.0	36,441	54.8%
Licensing Dropped	The parent is under one or more licenses, but the child has no specified license	apache-2.0 → UNDECLARED	19,435	29.2%
Licensing Added	The child is under one or more licenses, but the parent has no specified license	UNDECLARED → apache-2.0	10,548	15.9%
License(s) Added	In a multi-licensing scenario, the child retains all of its parent’s licenses and adds one or more additional licenses.	apache-2.0 → apache-2.0   mit	28	0.04%
License(s) Removed	In a multi-licensing scenario, the child is no longer under one or more licenses that apply to its parent, but is still licensed and adds no additional licenses.	apache-2.0   bsd-3-clause → bsd-3-clause	8	0.01%
<b>Total Changes</b>	-	-	<b>66,460</b>	<b>100.0%</b>

access to more data than was originally pulled for the rest of the dataset described in this paper, we conducted a separate smaller mining on February 1, 2025. Specifically, we pulled the top ten thousand models (sorted by downloads) that reported they were under the “Other” license. We acknowledge that the results of this subsequent mining may not perfectly align with the dataset presented in the rest of this work, but we believe it is worthwhile to explore the phenomenon of the “Other” license in more depth.

Table 18 shows the top 5 specific licenses that were reported in the CARDATA attribute. Over a third of models (36%) did not provide additional licensing information in their CARDATA. Three percent of models are actually under the LLAMA3 license, which is in the recognized license list on Hugging Face [35]. As mentioned before, this suggests that either these models were uploaded before LLAMA3 was added to the official list and have not been updated or that the model owners are unfamiliar with the licenses available in the official list. In either case, feasible tooling solutions can be readily imagined to resolve these problems.

There is a further lack of standardization in the licenses reported as “Other.” For example, 206 models report DEEPSEEK and an additional 85 report the DEEPSEEK-LICENSE, though these are presumably the same license. As previously mentioned, 269 models erroneously report LLAMA3 as their “Other” license, but an additional 27 report being under the LLAMA-3 license. We see 110 models under the GEMMA-TERMS-OF-USE, even though the proper identifier for this model is just GEMMA [35]. All of this further underscores the problems inherent in providing developers with a free-text field instead of assisted tool support that suggests known licenses.

### 7.3 Parent/child license differences

Our dataset contains 274,104 distinct parent/child relationships. We observed 66,460 instances (24%) where the licensing of a child model (*i.e.*, derivative model) was different from that of its parent (*i.e.*, base model). Table 19 provides an overview of the types of differences observed. In nearly a third of cases, child models specified no licensing information despite information being available for their parent(s), leading to a potential compliance issue. We also observe that once

Table 20. Most common parent/child license differences

Parent License	Child License	Count	Percentage
cc-by-nc-4.0	apache-2.0	11,001	30.19%
apache-2.0	cc-by-nc-4.0	7,427	20.38%
cc-by-4.0	cc-by-nc-4.0	2,626	7.21%
other	llama3	1,720	4.72%
cc-by-nc-4.0	cc-by-4.0	1,444	3.96%
other	apache-2.0	1,348	3.70%
apache-2.0	mit	1,241	3.41%
cc-by-nc-nd-4.0	cc-by-nc-4.0	1,180	3.24%
llama3	other	1,083	2.97%
apache-2.0	cc-by-4.0	782	2.15%

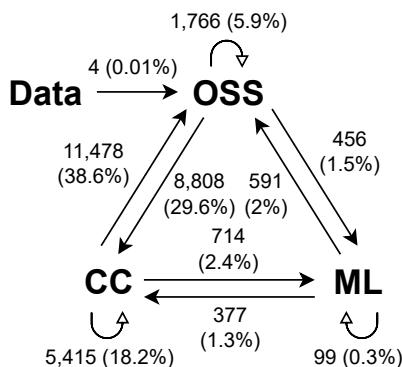


Fig. 4. Parent/child license differences across categories (Total differences: 29,708)

license information has been dropped, it is unlikely to be restored by later links in the chain, with license restoration behavior observed in only 15.9% of instances. A complete shift in licensing between parent and child was seen in 54.8% of cases (*i.e.*, no licenses were shared between the two). Of these shifts, 18.5% involved the “Unknown” or “Other” license category. Table 20 shows the top ten such shifts, and Figure 4 provides more information on how licensing decisions shifted across categories between parents and their children (*e.g.*, a difference from CC-BY-4.0 to MIT), as well as how licensing decisions shifted within categories (*e.g.*, a difference from MIT to APACHE-2.0). While ML-specific licenses seem to be a good fit for models, model owners may be opting for licenses with which they are more familiar and comfortable (OSS for software developers and CC for data scientists). We observe that most shifts occur between the OSS and CC categories (68.3%), perhaps suggesting some tension between the licensing preferences of those with data science and software development backgrounds. Additionally, depending on the specifics of the situation and the licenses involved, these shifts can also potentially introduce license incompatibilities, such as dropping non-commercial requirements or failing to apply the requirements of a copyleft license.

#### 7.4 Dataset license and model license combinations

The licenses of datasets that are used during training can have an impact on the licensing decisions for the final ML model. Here we consider the pairings of dataset licenses and the resultant model licenses. There are 623 distinct model/dataset license combinations within our dataset across 43,455 such pairings; however, since we (and Hugging Face) aggregated all “Other” licenses under one

Table 21. Model license and dataset license pairs

<b>Model License</b>	<b>Dataset License</b>	<b>Count</b>	<b>Percentage</b>
apache-2.0	other	8196	18.9%
apache-2.0	cc0-1.0	4621	10.6%
mit	mit	4479	10.3%
apache-2.0	apache-2.0	3916	9.0%
apache-2.0	unknown	2993	6.9%
apache-2.0	mit	2734	6.3%
mit	other	2290	5.3%
apache-2.0	cc-by-4.0	2076	4.8%
apache-2.0	cc-by-sa-4.0	1934	4.5%
mit	apache-2.0	1739	4.0%

label, these numbers likely overestimate the consistency in the space. (The top ten most frequent can be found in Table 21.) The license of the model exactly matches the license of at least one dataset it was trained on in only 41 of the 623 combinations. The most common combination is a model licensed under the APACHE-2.0 license and a dataset under a custom or “Other” license, with 11,731 (27%) pairs involving a dataset under a custom or “Other” license. This can make using these datasets and models problematic since retrieving and evaluating the licensing terms can be difficult and time-consuming.

## 7.5 Additional licensing findings

**7.5.1 Multi-licensing.** A few models (188) and datasets (104) were released under more than one license. Multi-licensing is an existing phenomenon in the world of open-source software but is made more complex in the ML model context by the presence of novel license combinations that are not yet well understood. Examples include the 48 models we observed multi-licensed under APACHE-2.0 and CC-BY-NC-4.0 as well as models under both OSS and ML licenses, such as MIT and OPENRAIL. We observe examples of such multilicensing in the most declared datasets in Table 9. We also observed datasets that were released under as many as six distinct licenses. The Hugging Face metadata provides no way of determining the relationship among these licenses, which may be an AND or an OR, or even indicate, as observed in Sections 8.4.2 and 9.3, that portions of the dataset are under different licenses. Regardless, this distinction is critical to determining compliance. For example, if a user who uploaded a LLAMA2 derivative model chose to make it available with an OR relationship between the APACHE-2.0 and LLAMA2 licenses, this would contradict the exclusive LLAMA2 licensing of its base model, resulting in a violation that could propagate to models further down the chain. While we observed only a few instances of multi-licensing, such activity motivates the need to understand the interplay between the various license classes and for Hugging Face to supply a standardized way to specify the intended relationship between licenses.

As a case study, we consider the most common example of multi-licensing found across models: a combination of the APACHE-2.0 and CC-BY-NC-4.0 licenses. Depending on the relationship between these two licenses and the reasoning for including them, the combination could represent potential compliance challenges. To explore this, we conduct a manual analysis of MRADER-MACHER/ NSFW\_DPO\_NOROMAID-7B-MISTRAL-7B-INSTRUCT-v0.1-GGUF, the model under this licensing scheme with the most downloads. This model is a quantization of MAZIYARPANAHI/ NSFW\_DPO\_NOROMAID-7B-MISTRAL-7B-INSTRUCT-v0.1, which is also dual-licensed in the same way. The model card for MAZIYARPANAHI/NSFW\_DPO\_NOROMAID-7B-MISTRAL-7B-INSTRUCT-v0.1 describes it as a merge of two additional models. The first, MISTRALAI/MISTRAL-7B-INSTRUCT-v0.1,

is under the APACHE-2.0 license, and the second, ATHIRDPATH/NSFW\_DPO\_NOROMAID-7B, is under the CC-BY-NC-4.0 license. Following the supply chain back further, we observe that the dataset that ATHIRDPATH/ NSFW\_DPO\_NOROMAID-7B was trained on was also licensed under CC-BY-NC-4.0. MISTRALAI/MISTRAL-7B-INSTRUCT-v0.1 is a finetune of MISTRALAI/MISTRAL-7B-v0.1, but there is no record of the data used for training. From all of this we see that the licensing relationship in this multi-licensing example is likely intended to be an AND (*i.e.*, that the model is released under the terms of both the APACHE-2.0 and CC-BY-NC-4.0 licenses). However, if other model trainers were to interpret the relationship as an OR (*i.e.*, the model is licensed such that creators of derivative works can freely choose which licensing terms, either the APACHE-2.0 or CC-BY-NC-4.0 license, to follow), it is possible that the non-commercial limitation imposed by the CC-BY-NC-4.0 license could be dropped and non-compliance could be introduced. We also note that here, instead of finding a single suitable license for MAZIYARPANAHI/NSFW\_DPO\_NOROMAID-7B-MISTRAL-7B-INSTRUCT-v0.1, the model owner opted to just include the licenses of both dependencies. While not observed in this case, it is possible that this strategy could fail in those instances where dependency licenses conflict.

**7.5.2 Naming requirements.** Some licenses, like LLAMA3, impose a generational limitation [73] that specifies naming requirements for derivative works. According to the license, models that are built on a model under this license must have names prefixed with “llama3” [82]. In our dataset, there are 3,335 models licensed under LLAMA3. However, of these, only 473 (14.2%) models correctly have names beginning with “llama3” as required by the license. Only 768 (23%) models even contain the term “llama3” in their name. Indeed, just 2,421 (72.6%) models contain the term “llama” at all, and 384 (11.5%) only contain the string “l3.” The problem is likely even worse in practice, as these statistics only include models that documented that they were under the LLAMA3 license.

**7.5.3 Attribution.** As was observed during the manual model card analysis described in Section 4.3, attribution appears to be important to model owners regardless of licensing status. Rather than requesting or requiring attribution through a license, models without a license suggested the desire for some form of attribution (29/84), typically in the form of citations. In fact, 83,628 models (11.1%) mined using the Hugging Face API included a link to a paper on arxiv [10] in their tags, thereby providing a means of citing the work behind the model. In our manual analysis of model cards, we observed that the model owner was typically also affiliated with (*e.g.*, an author on) the linked research paper. This desire for recognition is reflected by the top licenses in the space, namely MIT and APACHE-2.0, each of which specifies attribution requirements. (A similar historical example is the evolution of Creative Commons licenses in the early 2000s, where the overwhelming preference for attribution led to its becoming a default term [49].) Given the research community’s adoption of Hugging Face, it may be worthwhile to add dedicated metadata fields to facilitate the standardization of citation information.

## 7.6 Real World Potential Incompatibilities

Here, we go beyond the parent/child license differences explored in Section 7.3 and examine some observed potential incompatibilities. We note that the list of compatibility issues is not exhaustive, since there are many unresolved legal questions concerning the application of certain terms. We report the two most frequently observed issues in our dataset. While other issues were present, each only represented a few dozen examples in the entire ecosystem, so we exclude them from this section. We also exclude from this section any mappings that involve “Other,” “Unknown,” or undeclared licenses.

**7.6.1 Dropping of non-commercial requirements.** Of the total parent/child license differences, 30.19% involved a parent under CC-BY-NC-4.0 and a child licensed under APACHE-2.0. An additional 3.96%

involved a difference between the parent license cc-BY-NC-4.0 and the child license cc-BY-4.0. In both cases, the “nc” in the parent license indicates that the work can not be used for commercial purposes. The APACHE-2.0 license provides no such limitation, thus introducing a potential conflict. A change in requirements of such a nature could potentially lead to a license violation. Consider a situation in which a developer releases a component under cc-BY-NC-4.0, but a later developer fine-tunes or packages it under the MIT license. A user might seek to use this MIT-licensed version of this component in a commercial product, thinking the MIT license applies, though in the process they violate the non-commercial requirement of the cc-BY-NC-4.0 license under which the component was originally made available. In this sense, we may observe something akin to unintentional (or perhaps intentional) license laundering [77].

**7.6.2 Violation of no derivatives term.** Of the total parent/child license differences, 3.24% featured the parent license cc-BY-NC-ND-4.0 and the child license cc-BY-NC-4.0. The only difference between the two is the dropped “nd” condition, which indicates that no derivatives based on the licensed work are permitted. This means that if we consider a fine-tuned model to be a derivative of its base model, all 1,180 of these instances represent an example of noncompliance.

## 7.7 Summary

Overall, licensing on Hugging Face presents a number of unique challenges. While traditional open-source software licenses remain popular among ML components, the introduction of ML- and data-specific licenses raise novel questions regarding the compatibility of different licenses, and, as such, different components. The “Unknown” and “Other” licenses complicate the matter of compliance further by obscuring licensing information. To illustrate the specific difficulties that may arise, we gathered classes of potential licensing issues currently present in models on Hugging Face.

## 8 CASE STUDY: TOP MODELS

To better understand and illustrate the supply chain structure and challenges described in this section, we present a detailed analysis of the top 100 most downloaded models and the top 100 most liked models from our dataset. As described in Section 7.2.1, we suspect that model downloads can be used as a proxy for documentation quality.

### 8.1 Sample construction

Among the 100 top downloaded and 100 top liked models, there were 187 unique models. The 13 overlapping models were mostly uploaded by well-known technology companies, including OpenAI, Meta, StabilityAI, Google, and Microsoft. The only exception was the ALL-MINIILM-L6-v2 model uploaded by sentence-transformers. These 187 models collectively yielded 1,465,884,673 downloads, or 79.7% of all 1,838,905,161 downloads from the entire dataset of 760,460 models.

Forty-five (24%) are TEXT-GENERATION models, 28 (15%) are TEXT-TO-IMAGE models, and 17 (9%) are TEXT-CLASSIFICATION models. Other common model types included FILL-MASK (7%), SPEECH RECOGNITION (6%), and SENTENCE-SIMILARITY (5%). We also note that 21 (11%) models did not specify the pipeline (*i.e.*, a high-level abstraction that simplifies using models for common tasks, roughly corresponding to the model’s intended task [39]), revealing documentation shortcomings even for some popular models.

By far, the most common libraries (*i.e.*, open-source tools available through Hugging Face that support ML training [37]) used were TRANSFORMERS (121) and SENTENCE-TRANSFORMERS (11). The DIFFUSERS library was the next most common with 28 instances. Twelve models (6.4%) did not specify in their documentation any libraries.

Ninety-three unique model owners are represented in the top 187 models, including organizations like StabilityAI, Google, Meta, Microsoft, and OpenAI. In fact, 91 of the 187 top models (48.6%) are owned by just 15 model uploaders (without combining owners such as meta-llama, facebook, and facebookkai into one entity), suggesting that a small handful of players dominate the landscape. Sixty-three model uploaders (67.7%) have only one model in the top 187 models, suggesting long-tail behavior similar to that observed with respect to the dataset as a whole in Section 4.4.3.

Of the 187 models, 148 models (79%) are relied on by at least one other model available on Hugging Face (*i.e.*, they are base models). When considering the supply chain graph discussed in Section 6, we see that the 148 base models collectively have 43,976 outgoing edges, with a mean out-degree of 235 and a standard deviation of 665. Twenty-one models have an out-degree of one, and twelve have an out-degree of two. In total, 43,874 distinct models depend on this set of 148 models, further demonstrating their relevance to the ecosystem.

## 8.2 Documented base models

Four models (2.1%) in the top 187 declare a base model of their own (*i.e.*, they depend on another model). Given that many of the models in our set are foundation models, we would not expect them to be built on other publicly available models. Only three distinct base models were declared in the set, two of which were contained in the top 187 models themselves (MISTRALAI/MISTRAL-7B-v0.1 and XLM-ROBERTA-BASE). Despite the dearth of base model declarations, 8 models showed evidence of quantization (*i.e.*, completed the “quantization\_config” or “quantized\_by” fields in its metadata or included a file containing the substring “quant” in the repository) and were confirmed to be quantizations through manual analysis. However, none of these quantized models declared a base model in the appropriate metadata field (*e.g.*, CARDATA/basemodel), yet by virtue of being quantizations, these models must depend on a base model. This example illustrates the ongoing need for more thorough documentation and declaration of base models on Hugging Face.

In general, the supply chain of models in the top 187 is shallow. Few models declared a base model, and those that did relied on a single foundation model with no further dependencies. There were no instances in which a popular model relied on a less popular one. There were two parent/child license differences, a switch from OPENRAIL++ to CREATIVEML-OPENRAIL-M and a switch from APACHE-2.0 to MIT, but neither of these are likely to result in compliance challenges, given the permissive natures of the licenses involved.

## 8.3 Model licensing analysis

Next, we explore the presence of licensing information for models within our 187-model sample. Of that sample, only 165 provided any sort of licensing information, and of those, 80% had licensing information in both their tags and in their CARDATA field, 16% had licensing information only in their tags, and no models had licensing information exclusively in their CARDATA. This further reinforces that when considering completeness, the tags are the best source to mine. However, while the tags seem to be the most complete source of licensing information with respect to determining what license is declared, the CARDATA field, when completed properly, includes more specific information, such as external links to license files. Twenty-six models (16%) had a LICENSE file included in their repository. Of these, twenty-two models (84.6%) also had licensing information available in both their tags and CARDATA fields, and four models reported licensing information exclusively through the LICENSE file.

Conversely, there were twenty-four models (13%) that did not include any licensing information in their tags, CARDATA field, or a LICENSE file. We were able to determine the license for only two of these models (1%) through manual analysis. Six models (25%) included text in their documentation that provided terms, conditions, or guidance on derivative uses but that was not explicitly presented

Table 22. Licensing information location in Top 187 models

Location	Count	Percentage of licensed (n=165)
Tags and CARDATA	132	80%
Tags only	27	16%
LICENSE file only	4	2%
Model card text only	2	1%
No licensing information	22	-

as licensing information. (One of these six models stated that the model was open-source, but provided no specifics.) Eight of the twenty-four models (33%) provided a BibTeX citation, suggesting that the model owners either desire credit for their model or that they want to ensure that the original trainers receive credit.

Lastly, we note that in three cases licensing information was found only in a seemingly related GitHub repository, but it was often unclear whether the license applied to the model weights, for the code that runs/loads them, or both.

Ultimately, most model licenses in our sample are specified in the tags, CARDATA, or in a LICENSE file. Table 22 summarizes where licensing information was found for the models in this dataset. We note that examining the tags alone captures 96 percent of license declaration information, where such information exists.

Discrepancies between information sources were rare, but did occur. There were no license discrepancies encountered between the tags and the CARDATA fields. However, when comparing the license information available in the tags with included LICENSE files, six potential discrepancies were observed. One model reported in its tags to be under the CREATIVE-OPENRAIL-M, but provided a completely custom LICENSE file. There were four instances where models declared being licensed under OPENRAIL++ in their tags, but their LICENSE files were for CREATIVEML OPEN RAIL++-M, which is a subset of OPEN RAIL++. Lastly, the model STABILITYAI/STABLE-VIDEO-DIFFUSION-IMG2VID-XT was reportedly released under something called the stable-video-diffusion-community, but in reality, the LICENSE file was a verbatim copy of the STABILITY AI COMMUNITY LICENSE AGREEMENT (referred to as SAI-NC-COMMUNITY elsewhere).

Overall, we observed that the most popular models are likely to be permissively licensed. Specifically 66 of the licensed models (40%) were under the APACHE-2.0 license, with an additional 35 models (21%) made available under the MIT license. The next largest group, with 21 models (13%), were labeled with an “Other” license, explored in greater detail in Section 7.2. Thirteen models (8%) were under the CREATIVEML-OPENRAIL-M license, and six (4%) were under OPENRAIL++. Other licenses with at least two instances included LLAMA2 (5), CC-BY-NC-4.0 (5), LLAMA3 (3), OPENRAIL (2), GEMMA (2), BSD-3-CLAUSE (2), and CC-BY-4.0 (2). None of the most common models were declared to be under more than one license.

Only one model in the top models was licensed as “Unknown.” This appears to be an error, however, since there is a form at the top of the model card requesting the user to “Acknowledge license to access the repository,” clarifying that the model is under the FALCON-180B TII license. As such, the “Other” license option would likely have been more apt here.

Inversely, the OPENASSISTANT/OASST-SFT-6-LLAMA-30B-XOR model is reportedly under an “Other” license, but no additional licensing information is provided, potentially making the “Unknown” option more apt. Some legal guidance is provided however: “Due to the license attached to LLaMA models by Meta AI it is not possible to directly distribute LLaMA-based models. Instead we provide XOR weights for the OA models.”

The other twenty models under an “Other” license represent issuance under custom and vanity licenses. Four appear to be modifications of more popular licenses such as CREATIVEML OPEN RAIL-M. The others are unique licenses presumably written by the model owner or their organization. This suggests that model trainers are unaware of existing licenses, do not want to invest time researching the terms of existing licenses, or believe that existing licenses do not cover all terms, conditions, and use cases envisioned by the model trainers. Future work on the software engineering front will need to consider tooling solutions to help model trainers choose licenses that fit their specific needs. If it is the case that current licenses are not up to the task, it may be necessary for legal professionals and researchers to participate in drafting more robust and applicable licenses.

## 8.4 Documented datasets used

**8.4.1 Overview.** Only 56 (30%) of models in the sample record at least one training dataset in their metadata. While large technology companies may conceal training sources to mitigate the potential of legal liability [70], this is still a rate three times higher than the 10% observed across the ecosystem at large (Section 6.2). All 56 models recorded dataset information in both the tags and in the CARDATA attribute. A total of 107 datasets were declared across the sample of 187 models, though we could only verify the presence of 96 of these on Hugging Face. (The remaining eleven resulted in a 404 error when searching on Hugging Face.) Thirty-three models reported relying on only a single dataset, but the average was 3.4 with a standard deviation of 5. Three models recorded relying on as many as 21 datasets.

Even this small set highlights several challenges in uniquely identifying datasets, such as three datasets, all belonging to the same account, that had their unique identifiers changed when the dataset owner updated their profile name from *ehartford* to *cognitivecomputations*. Two additional datasets from *embedding-data* also had their names changed. Without Hugging Face maintaining a mapping between the old and new human-readable IDs, it would be extremely difficult to correctly determine these dependencies. Other ambiguities arise from the unstandardized nature of some declarations. For example, it is unclear whether the reference to “*wikihow*” by some of the top models refers to the WANGWILLIAMYANG/WIKIHOW dataset available on Hugging Face, some other dataset that compiles/aggregates information from wikiHow, or that the model was trained by scraping the wikiHow website. We explore the problems with identifying datasets further in Section 5.5.

### 8.4.2 Dataset licenses.

Unlicensed datasets: Of the 96 observed and mappable datasets used by the top models, twelve (12%) did not explicitly declare any licensing information in their tags. However, six of these datasets did declare at least some licensing information in other ways. For example, FLAX-SENTENCE-EMBEDDINGS/STACKEXCHANGE\_XML points users to an *arxiv.org* link that includes more specific licensing information for the dataset, including steps for attribution. Two datasets (KRESNIK/ZEROTH\_KOREAN and TIMDETTMERS/OPENASSISTANT-GUANACO) included licensing details in the text of their READMEs. Another dataset, OPENCHAT/OPENCHAT\_SHAREGPT4\_DATASET, provided a link to a GitHub repository that did include licensing information. However, the license in the repository is only explicitly stated to cover the codebase, not the dataset, though this may be implied. Lastly, CEREBRAS/SLIMPAJAMA-627B and TOGETHERCOMPUTER/REDPAJAMA-DATA-1T were aggregated datasets that pulled from a variety of other datasets. Their READMEs include links to the included datasets, leaving the user to determine the appropriate license based on which portions of the dataset are used. These examples highlight how despite the ability to automatically detect licenses in many cases, a human investigator may still need to perform manual analysis in others. Future work may investigate the usefulness of LLMs for this task.

**“Other” licenses:** Ten datasets declared that they were released under an “Other” license. One of these was multi-licensed with other, more standard licenses, and in 5 of these cases, the datasets were aggregates of other datasets or source code files under various licenses. Hence, the “Other” label was being used to convey that the end user will need to research the specific data points or data subsets used, which could be a tedious and time consuming task. Four datasets included licensing text/information in their READMEs. These were not standard license texts and in some cases were very minimal. For example, DAIR-AI/EMOTION states only, “The dataset should be used for educational and research purposes only.” Another dataset, JONDURBIN/AIROBOROS-2.2.1, provides text akin to a legal disclaimer: “Much (most) of the data was generated via gpt-4 API calls, which has a restriction in the ToS about ‘competing’ models. Please seek legal advice if you plan to build or use a model that includes this dataset in a commercial setting.” The final dataset, which was multi-licensed, declares Open Portion of the American National Corpus, which is a collection of texts and not a license. It is possible this was meant to indicate that the license(s) of this corpus apply to the dataset, or at least portions of it.

**The “Unknown” license:** Of the 96 observed and mappable datasets used by the top models, 8 (8%) were released under the “Unknown” license tag. Two of these were dataset aggregates where the different data points were made available under different licenses, such that choosing a single license for the whole dataset would be difficult, if not impossible. (This situation likely led other dataset owners to choose the “Other” license designation.) Three datasets had incomplete dataset cards/READMEs and did not include any licensing information. One model reiterated that the licensing of the released dataset was “Unknown” but included no elaboration. Two datasets stated that the licensing of the dataset depended on the licensing of its dependent data sources. One of these stated only that “[t]he University of Washington does not own the copyright of the questions and documents included in TriviaQA.” The other explained that “[t]he licensing status of the dataset hinges on the legal status of the Pushshift.io data which is unclear.” Despite using the “Unknown” license, presumably due to uncertainty regarding how their dataset should be licensed, six (75%) datasets still included citation information. Thus, similar to the “Other” license, the “Unknown” license is not a monolith, encompassing a number of different scenarios. Here, we observed its use for aggregates with multiple licenses, insufficient licensing information, or on its own without further explanation.

**Multi-licensing:** There were two datasets that reported being released under multiple licenses. The first of these, NYU-MLL/MULTI\_NLI, reported four licenses, including the “Other” license option. Clarification concerning how these licenses relate is found in the associated dataset card, which explains that different portions of the dataset are released under different licenses. The second example, LEGACY-DATASETS/WIKIPEDIA, reported two licenses, which also corresponded to portions of the dataset. In both of these examples, we see that multi-licensing has been used to describe a complex licensing scenario with different subsets of a dataset. This is contrary to the popular usage in open-source, where such a licensing scheme implies either an AND or an OR relationship between the licenses.

**License differences across models and datasets:** Given the difficulties in determining and interpreting the licensing for datasets that are multi-licensed or under the “Other” or “Unknown” license tags, we omit them from this analysis. To better understand relationships between the licenses of models and datasets, we present an analysis considering the remaining 107 model-dataset license pairs. Twenty-four pairs had licensing information that exactly matched (e.g., both the model and dataset were released under the MIT license). (We note that a perfect match is not required, nor does it necessarily indicate full compliance with all licensing terms.) Eighty-three pairs did not represent exact licensing matches. For example, one model was unlicensed despite relying

Table 23. Top 5 Model-Dataset license pairs in Top 187 models

Model License	Dataset License	Count
apache-2.0	mit	33
apache-2.0	cc-by-4.0	9
apache-2.0	cc0-1.0	0
apache-2.0	odc-by	8
apache-2.0	cc-by-sa-4.0	6

on seven datasets that had clearly defined licenses. Table 23 shows the most common pairings. Notably, the CC-BY-SA-4.0 and CC-BY-SA-3.0 licenses are restrictive and require derivative works to be released under the same license. Depending on whether models are considered derivatives of their training data, the pairings (APACHE-2.0, CC-BY-SA-4.0) and (APACHE-2.0, CC-BY-SA-3.0) may represent a compliance issue since the share-alike term for the dataset licenses would have been violated.

## 9 CASE STUDY: TOP DATASETS

Similarly to the above analysis of top models, we provide an analysis of the 100 most liked and 100 most downloaded datasets.

### 9.1 Sample construction

This sample of the top datasets on Hugging Face consists of 185 distinct datasets; fifteen datasets existed in both the top liked and the top downloaded. Across all 175,000 datasets in our full dataset, there are 49,466,329 total downloads. The 185 most popular datasets account for 41,994,158 of these, roughly 85%. This justifies that these models are worth further, more intensive investigation.

The datasets in the sample encompass 23 distinct tasks, including text generation (62), question-answering (35), fill-mask (13), text-classification (12), text2text-generation (9), multiple-choice (8), summarization (7), and others with four or fewer occurrences.

The sample contains 136 unique dataset owners. Only nine own more than two datasets in the sample, including Bigcode (8), AllenAI (5), EleutherAI (5), Mteb (5), Lighteval (4), HuggingFaceFW (3), HuggingFaceH4 (3), The Mozilla Foundation (3), and BAAI (3).

Compared with the model analysis discussed previously, we notice that the distribution of datasets seems to be more decentralized. Whereas the top 15 model uploaders owned 48.6% of the top models, the top 15 dataset uploaders only owned 27.6% of the top datasets.

### 9.2 Supply chain

We observe a number of different dependency relationships among datasets in the sample. One such relationship type involves transforming the data in one dataset into another format, such as the COHERE/WIKIPEDIA-2023-11-EMBED-MULTILINGUAL-v3 dataset, which is a chunked and embedded version of the WIKIMEDIA/WIKIPEDIA dataset. We also observe aggregates of full datasets, such as GARAGE-BAIND/OPEN-PLATYPUS and CEREBRAS/SLIMPJAMA-627B, or aggregates of such aggregates, such as TEKNIUM/OPENHERMES-2.5, which combines GARAGE-BAIND/OPEN-PLATYPUS (itself an aggregate) with various other datasets.

### 9.3 Licensing

**9.3.1 Presence of licensing information.** To determine the presence of licensing for the datasets in our collection, we used a combination of automated and manual processes. (We did not attempt to perform a legal analysis of any licensing decision, including whether any particular dataset

was protectable under copyright law.) First, we used the Hugging Face API to obtain licensing information from the dataset tags if it was available. For any datasets where we were unable to identify a license through this process, we engaged in a manual analysis of the dataset card. Forty-four datasets required such analysis; of these, 9 (20%) provided an exact license, while 14 (32%) offered some type of related licensing information, such as rules governing how individual portions of the dataset were licensed. Licensing information sometimes had to be inferred from other sources; for example, the license of one dataset (2%) could only be inferred from a linked GitHub repository, and 5 datasets (11%) explained that they were aggregates of other datasets, encouraging users to “Please refer to the licenses of the data subsets you use.” We were unable to access one dataset, which has presumably been removed from Hugging Face or made private, and we were unable to locate any licensing information for the remaining 28 datasets. Thus, in total, we obtained licensing information for 156 datasets (84%) in our collection, though this information was not always sufficient to derive the exact licensing terms associated with the usage of a dataset. We note that this is lower than the 91% observed in the most popular models. Several possible explanations exist for the lack of licensing information in datasets. For example, it might be the case that reconciling the licensing and copyrights of thousands of data points or even dozens of other datasets is outside the scope and expertise of most data scientists and model trainers. It might also be the case that datasets are often created with the end goal of training an AI model, and licensing/documenting them is not viewed as a priority by their creators.

**9.3.2 Common licenses.** Here, we look at the 131 datasets that had identifiable, non-“Unknown” licenses (including custom licenses). Thirty-six (27%) were released under MIT, 26 (20%) under APACHE-2.0, 11 (8%) under CC-BY-4.0, 10 (8%) under ODC-BY, 9 (7%) under CC-BY-SA-4.0, 8 (6%) under CC0-1.0, and 7 (5%) under CC-BY-NC-4.0. We observe that, for the most part, these numbers mirror those for the entire ecosystem (as seen in Table 14), with the primary exceptions being the omission of the OPENRAIL license and the inclusion on the ODC-BY license. We also note that a majority of the top datasets are permissively licensed.

**9.3.3 Analysis of the “Unknown” license.** Six datasets were tagged with the “Unknown” license, for various reasons. One, BAAI/COIG-PC, was a dataset aggregate that provided rules on determining which licensing terms applied to which portions of the dataset. An additional three included no clear licensing guidance, and, in fact, one hinted at violating the Terms of Service of a platform to create the dataset. The license for one dataset could be inferred from a linked GitHub repository, but the certainty is low. Lastly, the bookcorpus dataset, instead of offering simple, actionable usage limitations or requirements, instead included a long essay on the complications and ethics associated with mining data from a site (in this case smashwords) where the data authors are unaware that their material could be used for training machine learning models and have therefore not given consent for that usage. The dataset is still available on Hugging Face as of this writing, but the legality surrounding its use seems unclear even to the dataset owner.

**9.3.4 Analysis of the “Other” license.** The sample also includes 16 datasets tagged with the “Other” license. Of these, nine (56%) were aggregates that had individual datasets or data points licensed under differing terms. We discuss the implications of these composite datasets in Section 10.2. Four (25%) were under a custom license or non-standard licensing text, such as a short statement about the intended use (*e.g.*, “The dataset should be used for educational and research purposes only”) or conditional licenses (*e.g.*, “If the source data of LIMA has a stricter license than CC BY-NC-SA, the LIMA dataset follows the same. Otherwise, it follows the CC BY-NC-SA license.”). The latter example illustrates how the task of license compliance—or, indeed, the task of determining the licenses within the component itself—is often left to the user. Two additional datasets (12%) utilized

the “Other” tag, but we were unable to locate any additional licensing information, rendering them effectively unlicensed. Lastly, we observed one dataset under the “creative-commons-by-nc” license. This nonstandard form, not present in Hugging Face’s license list [35], is likely intended to refer “cc-by-nc-2.0,” “cc-by-nc-3.0,” or “cc-by-nc-4.0.” Without additional information, however, it is impossible to determine with certainty the exact intended license, further highlighting the limitations of using plain text fields and motivating the need for validation.

**9.3.5 Analysis of multi-licensed datasets.** Three datasets in the sample—WIKIMEDIA/WIKIPEDIA, SALESFORCE/WIKITEXT, and LEGACY-DATASETS/WIKIPEDIA, all of which are composed of text pulled from Wikipedia—reported more than one license in their tags, a combination of the CC-BY-SA-3.0 and GFDL licenses. We note that user-written text that is submitted to Wikipedia is generally under one of these two licenses as well. This example illustrates how multi-licensing schemes can attempt to capture the complexities of the licensing relationships of the individual data points that make up a dataset: both WIKIMEDIA/WIKIPEDIA and LEGACY-DATASETS/WIKIPEDIA provide nuance with statements such as, “Some text may be available only under the Creative Commons license; see their Terms of Use for details. Text written by some authors may be released under additional licenses or into the public domain” and “Some text has been imported only under CC BY-SA and CC BY-SA-compatible license[s] and cannot be reused under GFDL; such text will be identified on the page footer, in the page history, or on the discussion page of the article that utilizes the text.” Lastly, although SALESFORCE/WIKITEXT tags both licenses, in the body of its model card it simply states, “The dataset is available under the Creative Commons Attribution-ShareAlike License (CC BY-SA 4.0),” excluding any mention of the GFDL. In addition to these examples, there were three other datasets that provided no licensing information in their tags but were revealed to be multi-licensed through manual analysis: UONLP/CULTURAX, OPEN-WEB-MATH/OPEN-WEB-MATH, and SKYWORK/SKYPILE-150B. All three explicitly stated that the datasets were released under the conditions of both licenses, representing an AND relationship (as described in Section 7.5.1). Recall that Hugging Face does not provide a built-in way to specify such a relationship, necessitating this clarification from the models. As shown from these examples, multi-licensing information can sometimes be obscured in a way that limits the abilities of users to understand which licenses apply in what manner, which can open the door to noncompliance by dependents.

## 10 DISCUSSION AND IMPLICATIONS

In this section we discuss the implications of our findings and potential directions for future work.

### 10.1 License compliance in the ML supply chain

In Section 6, we showed that the structure of the ML supply chain on Hugging Face consists primarily of relatively short component chains. This presents a radically different landscape from the traditional software supply chain, in which a single piece of software can depend on potentially hundreds or even thousands of components. However, just because fewer dependencies are declared does not mean that ensuring license compliance is any easier in this ecosystem. In fact, as we observed, licensing information is often missing (Section 7) and there can be licensing differences between parent and child models in a model’s lineage (Section 7.3). Compounded with our observation that datasets may also have unclear licensing (Section 7.1) and may go undeclared by models (Section 6.2), the state of documentation on Hugging Face appears to be counterproductive to successful license compliance.

The task of managing different licenses across a model’s components presents a novel challenge due to the advent of ML-focused licenses and a heavy reliance on CC licenses. While previous work has investigated compatibility between various traditional software licenses [68, 76], new licenses,

such as the OPENRAIL family and the LLAMA licenses, require analysis to understand how they interact with existing licenses and with each other. Additionally, ML-focused licenses also indicate legitimate uses for the model and any derivatives, further separating them from the traditional open-source definition. Proponents of OPENRAIL argue that existing open-source licenses are “ill-adapted” to handle ML components’ differences from source code, particularly with respect to the responsible use of such components [44]. This necessitates future work to provide a detailed analysis of software licenses with respect to ML components, including an analysis of these licenses and the ways in which licenses interact with ML components differently from traditional software.

It also becomes crucial to educate model owners and users on the terms associated with licenses. For example, we observed that the simple naming requirements imposed by the LLAMA3 license were followed in only 14.2% of cases (Section 7.5.2). This suggests either extensive deliberate noncompliance or, more likely, ignorance of the relevant license terms. To concretely identify the cause and provide potential mitigations, future work should investigate developers’ views and understanding of licenses in the ML supply chain.

It also remains unclear, particularly with the relationship between datasets and models, how exactly license terms, such as GPL’s copyleft, should be applied [54]. To what extent should a model be considered a derivative work of the dataset(s) on which it was trained? More concretely, if a model trainer trains a model on a dataset governed by a copyleft license and then distributes the model, but not the dataset, has the trainer violated the dataset license? More fundamentally, to what extent are ML datasets protected by copyright law in various jurisdictions? In other words, is a particular selection and arrangement of data protected by copyright, or does copyright attach only at the level of individual data, if at all? These questions and many others will likely need to be resolved by legal teams, regulators, and possibly the courts.

We observed challenges presented by licensing datasets that contain data points under different licenses (Sections 9.3.3 and 9.3.4). While these challenges are not new [75, 77], future research and the Hugging Face development team should investigate more intuitive ways of conveying that the licensing of a dataset depends on the specific portions used rather than allowing users to obscure this information by the use of an “Other” label, instead reserving the “Other” label for custom or non-standard licenses. Relatedly, dataset and model repositories should encourage model/dataset uploaders to select from a list of already existing licenses rather than using “Other” in order to avoid rampant license proliferation. If custom license text is required, it should be within a LICENSE file so that it is easy for users to locate and access.

Lastly, in Sections 7.5.1, 8.4.2, and 9.3.5, we also observed instances of ambiguity introduced through multi-licensing. Determining the intended relationship among the multiple licenses assigned by the model/dataset owner can be a difficult task, and different pitfalls come with the assumption of an OR versus an AND relationship. Model hubs should include standardized ways for model/dataset owners to express these relationships and educate users on why they might want to use a multi-licensing scheme.

## 10.2 Dataset aggregates

As discussed in Sections 9.3.3 and 9.3.4, some datasets on Hugging Face were aggregates or composites of other datasets. In these situations, the uploader would leave license identification to the end user, saying something to the effect of “Look at the licenses for the different subsets of this dataset.” In addition to raising potential compliance questions, the lack of standardization in the way that this aggregate nature was conveyed potentially presents additional challenges to informed usage. We observed model uploaders using the “Unknown” and “Other” license options to refer to this composite nature. Other uploaders opted to leave their dataset unlicensed in the tags and elaborated in the body of the dataset card that the terms of the individual licenses needed to be

complied with. Still others opted to declare multiple licenses in their tags, not to indicate an AND or an OR relationship but rather to indicate that different licenses applied to different subparts of the dataset (Section 9.3.5). Without an additional field to allow owners to describe this type of licensing situation more precisely, this structure is likely to contribute to developers' difficulty with understanding licensing tasks, a difficulty that previous work has explored [8].

The purpose of these dataset aggregates would seem to be assembling large quantities of training data for machine learning models and making them easily accessible. Additional problems arise if the licenses contained within an aggregate conflict. For example, NGUHA/LEGALBENCH is composed of datasets that are under two different copyleft CC licenses (ATTRIBUTION-SHAREALIKE 4.0 INTERNATIONAL and ATTRIBUTION-NONCOMMERCIAL-SHAREALIKE 4.0 INTERNATIONAL). It is not possible to meet the terms of both licenses, meaning that this dataset composite and models trained on this dataset risk being deemed noncompliant. This also poses risks for any end user, who might not have the motivation or the knowledge to determine compliance, raising nonfunctional ethical considerations that are deserving of attention by researchers.

### 10.3 Potential difficulties in using model cards as ML/AIBOMs

Amidst the growing push for better transparency and security in software through Software Bills of Materials (SBOMs) [96, 116], calls have been made for similar BOMs for ML components [12, 13, 96, 116]. These documents serve as inventories of all components within a piece of software and can document a variety of the software's traits, including dependencies, licensing, and security. Distinct ML/AI Bills of Materials (ML/AIBOMs) may be necessary to address such components' different inputs, security concerns (such as model poisoning), and ethical considerations [17, 78, 116, 117]. The SPDX [95] Working Group, operating under the auspices of the Linux Foundation, is developing guidelines and a proposed standard for ML/AIBOMs [15] that will complement the SPDX ISO Standard ISO/IEC 5962:2021, which describes the use of SBOMs to document the components used in creating a software system [58]. CycloneDX [29] is also working on support for ML/AIBOMs [30].

Prior work has also suggested that model and data cards can serve as an ML/AIBOM [96], or that the information they provide can assist with their creation [93, 117], but our work suggests that these tools are not yet robust enough to serve this purpose. As we highlight in Section 5, the information provided by model cards is often missing key elements, including datasets that the models were trained on. This immediately obviates many of the benefits of ML/AIBOMs, including understanding licensing obligations that might be associated with that dataset, being aware of potential model poisoning attacks, and providing the ability to select models trained on ethically sourced data. Additionally, instances in which model cards are locked behind Terms of Service agreements or other restrictions (Section 5.1) could further limit their usefulness as ML/AIBOMs to consumers. Our work provides additional evidence that, in practice, model cards contain little actionable information, making them difficult to use as ML/AIBOMs [16] or as sources of information for producing an ML/AIBOM.

### 10.4 The need for follow up surveys and interviews

Our study raises many questions concerning what motivates developers when they make decisions concerning documentation, model reuse, and licensing. Conversations with Hugging Face developers through surveys and interviews can help to bridge this gap. There are numerous key questions such work should consider. What factors influence developers when choosing a license for their dataset/model and what do they believe the scope of that license to be (Section 7.1)? Why do developers so commonly opt to use an "Other" license (Section 7.1)? What motivates developers to multi-license a model/dataset and what is the intention behind the license interaction (Section 7.5.1)? Why do so few developers fine-tuning the LLAMA3 model comply with the naming

requirements imposed by the LLAMA3 license (Section 7.5.2)? What is the primary motivation for developers to write documentation for their models? For each of the various stakeholders in the space (model owners, fine-tuners, and users), which metadata fields do they perceive as most important? Why are some models listed as datasets (Section 5.6)? Answers to these questions will inform how future tools, practices, techniques, and educational materials should be developed to have maximal impact on the emerging ML community.

## 10.5 The addition of fields for recording model relationship information

As the relationships between ML models become more complex, it becomes increasingly important to keep a record of those relationships (fine-tuned, quantized, ensemble, distillation, etc.). This information is crucial not only for transparency and understanding how the model was created but also for license compliance. Just as in traditional software, licenses may have terms that govern the manner in which the component may be (re)used and distributed. For example, many generative models disallow using generated content to train a derivative model [90]. As noted in Section 5.7, Hugging Face provides no standardized way to express relationship information. This inability to specify relationships may have resulted in some model owners declaring models used in student-teacher relationships as datasets (Section 5.6). We suggest that model hubs provide a standardized, machine-readable way of conveying this type of information. Just as Hugging Face currently provides a standard list of architectures, it could also provide a curated list of relationship types.

## 10.6 The need to use unique identifiers

While unique identifiers already exist on Hugging Face, they are not currently being used by model/dataset owners. This can result in problems, particularly when trying to trace model/dataset provenance. Model hubs should encourage their users to use these existing unique identifiers, perhaps by including tools that facilitate the creation of model cards or the provision of metadata. For example, a developer could supply a colloquial name for a model hosted on the platform, such as the familiar owner/model convention, and the tool could automatically conduct a search and, where possible, map the provided name to the unique ID. This would immediately alert the developer to typos, remove ambiguities, and prevent the name drift issues we discussed in Sections 5.4 and 5.5. Furthermore, such a system could potentially allow for automatic updates (or request manual developer updates) when the information for downstream elements changes, thereby facilitating an ecosystem which is more supportive of supply chain management tasks and lifting the burden from downstream developers.

## 10.7 Differences between the ML supply chain and traditional software supply chain

Our findings highlight a notable difference between the ML supply chain and the traditional software supply chain. In many software ecosystems with dependency management, dependency information for components in the software supply chain is organized in manifest files such as `pom.xml`, `package.json`, or `requirements.txt`. These files can then be used to generate other dependency tracking documents like SBOMs and have also been used by forges like GitHub to build dependency graphs [1]. Notably, however, these manifest files have utility beyond dependency tracking. They are necessary for setting up a fresh installation of the software—that is, the list of dependencies that must be downloaded in order for the software to function properly. This is not the case with the information supplied by developers in model cards. Because erroneous mappings to datasets and base models do not leave the model in question unusable as a practical matter, it is therefore easier for typos and other mistakes to go unnoticed and thus uncorrected. This disconnect between correctness and functionality necessitates tooling that has similar functionality and

richness to dependency management tools for traditional software, as it leaves ample opportunity for errors without comparable means of checking for and correcting them.

### 10.8 Quality control for model hubs

Our analysis of the model documentation available on Hugging Face shows that the information that developers provide is often incomplete, malformed, or available only in human-readable text. We observed some documentation issues, such as the presence of duplicates in lists of datasets and base models (Section 5.7) or the declaration of a model as its own base (Section 5.5) that could be easily corrected by automated tools like linters. Additionally, an automated tool could also provide standardization for data fields to facilitate metadata analysis. For example, the variants for how one can specify that there are no datasets or base models ([], "", and NONE) should be standardized to a single type. We are not suggesting that model hubs should be responsible for curating the metadata of all models on their platforms, but where automated checks can efficiently be put into place and are easy to implement, they should be employed.

Potential solutions could closely resemble approaches to automatically check the quality of bug reports. Similarly to how tools can check for important components of a bug report (*e.g.*, expected behavior, steps to reproduce) [25], in theory, tools may be able to check for missing information in model cards. Some work is already emerging in this area, such as DocML [16] and AIMML [103]; however, more work needs to be done to understand what suggestions should be made to model owners, how to handle complex licensing scenarios, and how to best implement tools that assess ML documentation.

Since license metadata for models and datasets is entered manually, there is the possibility that typos are introduced. We encountered instances where the standard SPDX identifier CC-BY-SA-4.0 was mistakenly entered as cc-BY-SA 4.0 (*i.e.*, missing the final hyphen). While not a major issue, typos like these necessitate that tools perform data cleaning in addition to any work related to license compliance. Again, it would be preferable if Hugging Face provided a basic linter/validator for model/dataset cards and provided users with a drop-down/auto-complete from which to select popular licensing options, if this would be easy to implement.

### 10.9 Potential pitfalls when mining Hugging Face

The numerous documentation challenges that we discuss in Section 5 demonstrate that researchers relying on Hugging Face mining to collect information on the ML supply chain must proceed thoughtfully, with caution and care, to avoid potential pitfalls. Unlike source code forges like GitHub, it seems that some model owners may be uploading and publishing models as a by-product of their workflows [38, 40]. As such, we speculate that some models are made publicly available not because the owner intends them to be reused by the community but rather because the owner is using the resources of Hugging Face, Google Colab, or some other cloud provider [18]. This might explain why we observed so many models with the same names, such as MY\_AWESOME\_MODEL which is the example name used in the documentation for uploading a PyTorch model [40]. (See Section 5.4.) More research is required to understand how such tools might populate the supply chain with such “intermediate” models or those that are otherwise not intended for public consumption. To mitigate the risk such models pose, future studies should be mindful of the models they include in their analysis, erring toward those that are more popular. However, we caution researchers to be thoughtful in how they determine model popularity. In the context of the supply chain, we believe that downloads is the more appropriate metric, as opposed to likes. As shown in Table 6, the most liked “dataset” on Hugging Face is in reality a repository curating ChatGPT system prompts (*e.g.*, “Imagine you are an experience Ethereum developer...”). Additionally, the repository for LLLYASVIEL/SD\_CONTROL\_COLLECTION (with 1,574 likes, but 0 downloads) is actually a collection of various models assembled “for users

to download flexibly.” Future studies will be left to explore the prevalence of curated repositories hosted as models/datasets and the exact nature of potentially ad hoc developer workflows.

## 10.10 Recommendations for Stakeholders

Here, we offer concise and actionable recommendations for various stakeholders involved in the ML supply chain based on our results.

### 10.10.1 Model trainers and dataset curators.

- (1) When licensing models and datasets, use tags, the CARDATA attribute, and/or a LICENSE file rather than including licensing information in the README or via an external source.
- (2) When releasing a model or dataset under a multi-licensing scheme, clearly indicate how the licenses relate.
- (3) When choosing a license for models and datasets, where possible use common, pre-existing licenses, rather than drafting a new license or modifying the terms of an existing one.
- (4) When there is no choice but to draft a new license, make sure that the full license text is made available in a LICENSE file available on the model/dataset repository. Also ensure that licensing terms are unambiguous and comprehensive (e.g., more than “This model is open source”).
- (5) When using models and datasets, particularly for larger or commercial projects, verify that the declared license is consistent and compliant with other upstream sources.

### 10.10.2 Model and dataset sharing platforms.

- (1) Provide intuitive ways for dataset curators to specify the licenses of their work, particularly in the case of dataset aggregates. Simply using the “Other” or “Unknown” license options or declaring multiple licenses without clarification seems insufficient.
- (2) When designing documentation tools for model trainers and dataset curators, offer user-friendly options, such as multi-select widgets (with an “Other” option) and smart auto-complete text fields, instead of simple plain text fields. This is particularly the case for fields that have a finite set of recognizable values, such as licenses and pipelines.
- (3) Perform basic validation of model/dataset metadata, including checking for obvious typos, needless repetitions, or improper uses of fields.
- (4) Automatically include unique identifiers for models and datasets listed as dependencies when possible. Human-readable names for both models and datasets can change over time, but humans are also not well suited for working with the random strings of characters often associated with unique identifiers. When models/datasets are referenced in the metadata of other models/datasets, they should also be accompanied with a unique identifier that anchors the dependency, even if the human-readable name should change. For Hugging Face, this might mean automatically adding the internal model ID as part of the base\_model information in the CARDATA attribute.
- (5) Offer standardized, precise ways for model trainers to specify the dependency relationships between models (fine-tune, distillation, quantization, etc.), as these may have implications for various supply chain management activities, including licensing compliance.
- (6) For platforms that host both models and datasets, allow model trainers to specify where datasets are located with an external link if they are not available directly on the platform in order to make various provenance tasks significantly easier.

### 10.10.3 SE and Law Researchers.

- (1) Be aware and cautious of the limitations and quirks of mining data from Hugging Face. For example, the readily available “downloads” metadata field actually only reports downloads that have occurred within the last 30 days. Total downloads can be accessed on a per query basis by

providing additional arguments through the API, but it is not possible (as far as the authors can tell) to have the API return a list of models ordered by total downloads.

- (2) Choose proxy metrics carefully. As described in Section 10.9, when using model/dataset popularity/engagement as a proxy for quality or importance, be mindful that like count (similar to GitHub’s stars) may not be as reliable as the download count.
- (3) Understand that dependency information in the ML supply chain is more difficult to validate than dependency information in the traditional supply chain as described in Section 10.7.
- (4) Develop updated guidance and compliance matrices that can support stakeholders in the ML ecosystem as they work with various different licenses in a novel context.
- (5) Consider investigating developers’ views and understanding of licenses in the ML supply chain through surveys and interviews to better comprehend developers’ decision-making processes.
- (6) Know that not all models on Hugging Face were uploaded with the intent to publish. Instead, some may be incidental uploads that are a by-product of developer workflows through which they access cloud computing resources.

## 11 THREATS TO VALIDITY

**Construct Validity.** We rely on information that is available through the Hugging Face API in a machine-readable format. We addressed cases where information was missing from this source by applying Python’s request and BeautifulSoup libraries to retrieve missing information. However, this means that we likely missed licensing and dependency information that was presented in natural language. We measure and mitigate this risk with a review of the top 100 “unlicensed” models sorted by number of downloads, which showed that the majority did not, in fact, declare a license or declared a license in a non-standard or transitive way.

**Internal Validity.** We followed best practices in our mining and data cleaning approaches. Conclusions were drawn based on empirical evidence found in the data. We relied on tags applied to components to detect their licenses, but such information might be present in other, human-readable forms or through links to other resources, such as a research paper or another repository. The labeling of license categories is based on our understanding of the licenses. To mitigate bias in these labels, two researchers examined each license to confirm the label applied to it.

**External Validity.** The generalizability of our findings is restricted by our choice of Hugging Face, which might not be fully generalizable to all model-sharing platforms. It is possible that we missed other challenges and obstacles faced by other model hubs or that different model-sharing sites have different levels of documentation and dependency management techniques, given our assumption that highly downloaded models tend toward better documentation. We also observed that relatively few models and datasets declared licensing information, and models often did not declare any datasets. This implies that a great portion of the ecosystem may yet be undocumented. As such, our results may be biased toward the components that do provide documentation and may be less generalizable to those which do not. We attempt to mitigate this risk by providing a detailed analysis of top-rated components to capture the specific challenges facing components that are used frequently and are thus applicable to many users. Additionally, given that the field of AI is rapidly evolving and new models are created every day, the conclusions we reached might be different if we were to take an updated snapshot of the supply chain at some future point. The extent to which this factor might change our results is unknown, but we maintain that it is important to capture an understanding of this nascent supply chain so that future investigations can understand the ways in which the supply chain has changed over time.

## 12 CONCLUSION

In this work we conduct a comprehensive mining study and analysis of the ML supply chain on Hugging Face. We construct a reusable dataset of over 750,000 models and 175,000 datasets, which we make freely available in our replication package [97]. Using this data we create a supply chain graph that illustrates the relationships between ML models and the models they depend on, describe the notable features and properties of this supply chain, analyze the current state of documentation on Hugging Face, and consider the current state of licensing and license compliance in the ecosystem. We offer a rich discussion of the challenges that not just Hugging Face but any sizable model hub will encounter. Lastly, we offer concrete and actionable recommendations for various stakeholders involved with the ML supply chain that can be adopted to facilitate better supply chain management.

## DATA AVAILABILITY

We provide a replication package containing the raw mined data, our cleaned and normalized dataset, scripts for data cleaning, mining utilities, analysis scripts, and other data required for verifiability [97].

## 13 ACKNOWLEDGMENTS

This research has been supported in part by NSF grant CCF-2217733. Any opinions, findings, and conclusions expressed herein are the authors' and do not necessarily reflect those of the sponsors. A complete, detailed list of image attributions can be found at [97].

## REFERENCES

- [1] [n. d.]. About the dependency graph. <https://docs.github.com/en/code-security/supply-chain-security/understanding-your-software-supply-chain/about-the-dependency-graph>.
- [2] [n. d.]. Apache License. <https://opensource.org/license/apache-2-0>. Accessed: 2025-14-06.
- [3] [n. d.]. GNU General Public License version 3. <https://opensource.org/license/gpl-3-0/>. Accessed: 2023-14-09.
- [4] [n. d.]. MIT License. <https://opensource.org/license/mit/>. Accessed: 2023-14-09.
- [5] 2021. Copyright Registration of Computer Programs. <https://www.copyright.gov/circs/circ61.pdf>. Accessed: 2023-25-09.
- [6] 2021. U.S. Code Title 17, Section 106. <https://www.govinfo.gov/app/details/USCODE-2021-title17/USCODE-2021-title17-chap1-sec106/summary>. Accessed: 2023-25-09.
- [7] Frank Morales Aguilera. 2025. Navigating the AI Licensing Labyrinth: Truly Open vs. Restricted "Open-Weight" Models. <https://medium.com/ai-simplified-in-plain-english/navigating-the-ai-licensing-labyrinth-truly-open-vs-restricted-open-weight-models-89de5c2e649d>.
- [8] Daniel A Almeida, Gail C Murphy, Greg Wilson, and Michael Hoye. 2019. Investigating whether and how software developers understand open source software licensing. *Empirical Software Engineering* 24 (2019), 211–239.
- [9] Andersen v. Stability AI Ltd. [n. d.]. Andersen v. Stability AI Ltd. 3:23-cv-00201, (N.D. Cal.).
- [10] arxiv [n. d.]. arxiv. <https://arxiv.org/>.
- [11] Authors Guild v. OpenAI Inc. 2023. Authors Guild v. OpenAI Inc., 1:23-cv-08292, (S.D.N.Y.).
- [12] Iain Barclay, Alun Preece, Ian Taylor, Swapna Krishnakumar Radha, and Jarek Nabrzyski. 2022. Providing assurance and scrutability on shared data and machine learning models with verifiable credentials. *Concurrency and Computation: Practice and Experience* (2022), e6997.
- [13] Iain Barclay, Alun Preece, Ian Taylor, and Dinesh Verma. 2019. Towards traceability in data ecosystems using a bill of materials model. *arXiv* (2019).
- [14] beautifulsoup 2025. BeautifulSoup. <https://pypi.org/project/beautifulsoup4/>.
- [15] Karen Bennet, Gopi Krishnan Rajbahadur, Arthit Suriyawongkul, and Kate Stewart. 2024. A Comprehensive Guide to Creating AI and Dataset Bill of Materials. <https://spdx.dev/implementing-an-ai-bom/>
- [16] Avinash Bhat, Austin Coursey, Grace Hu, Sixian Li, Nadia Nahar, Shurui Zhou, Christian Kästner, and Jin LC Guo. 2023. Aspirations and practice of ML model documentation: Moving the needle with nudging and traceability. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–17.

- [17] Tingting Bi, Boming Xia, Zhenchang Xing, Qinghua Lu, and Liming Zhu. 2024. On the way to sboms: Investigating design issues and solutions in practice. *ACM Transactions on Software Engineering and Methodology* 33, 6 (2024), 1–25.
- [18] Jeff Boudier and Philipp Schmid. [n. d.]. Hugging Face and Google partner for open AI collaboration. <https://huggingface.co/blog/gcp-partnership>.
- [19] Blake Brittain. 2023. Authors sue Meta, Microsoft, Bloomberg in latest AI copyright clash. <https://www.reuters.com/legal/litigation/authors-sue-meta-microsoft-bloomberg-latest-ai-copyright-clash-2023-10-18/>.
- [20] Joel Castaño, Silverio Martínez-Fernández, and Xavier Franch. 2024. Lessons Learned from Mining the Hugging Face Repository. *arXiv preprint arXiv:2402.07323* (2024).
- [21] Joel Castaño, Silverio Martínez-Fernández, Xavier Franch, and Justus Bogner. 2023. Exploring the Carbon Footprint of Hugging Face’s ML Models: A Repository Mining Study. In *2023 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. IEEE, 1–12.
- [22] Joel Castaño, Martínez-Fernández Silverio, Xavier Franch, and Justus Bogner. 2024. Analyzing the evolution and maintenance of ML models on Hugging Face. In *Proceedings of the 21st International Conference on Mining Software Repositories*. 607–618.
- [23] cc-by-4.0 [n. d.]. Attribution 4.0 International. <https://creativecommons.org/licenses/by/4.0/>. Accessed: 2025-14-06.
- [24] cc-by-sa-4.0 [n. d.]. Attribution-ShareAlike 4.0 International. <https://creativecommons.org/licenses/by-sa/4.0/>. Accessed: 2025-14-06.
- [25] Oscar Chaparro, Jing Lu, Fiorella Zampetti, Laura Moreno, Massimiliano Di Penta, Andrian Marcus, Gabriele Bavota, and Vincent Ng. 2017. Detecting missing information in bug descriptions. In *Proceedings of the 2017 11th joint meeting on foundations of software engineering*. 396–407.
- [26] Kenneth Ward Church, Zeyu Chen, and Yanjun Ma. 2021. Emerging trends: A gentle introduction to fine-tuning. *Natural Language Engineering* 27, 6 (2021), 763–778.
- [27] Thomas Claburn. 2022. GPL legal battle: Vizio told by judge it will have to answer breach-of-contract claims. [https://www.theregister.com/2022/05/16/vizio\\_gpl\\_contract/](https://www.theregister.com/2022/05/16/vizio_gpl_contract/). Accessed: 2023-14-09.
- [28] Thomas Claburn. 2023. John Deere urged to surrender source code under GPL. [https://www.theregister.com/2023/03/17/john\\_deere\\_sfc\\_gpl/](https://www.theregister.com/2023/03/17/john_deere_sfc_gpl/). Accessed: 2023-14-09.
- [29] cyclonedx 2025. CycloneDX. <https://cyclonedx.org/>
- [30] cyclonedx 2025. Machine Learning Bill of Materials (ML-BOM). <https://cyclonedx.org/capabilities/mlbom/>
- [31] James C Davis, Purvish Jajal, Wenxin Jiang, Taylor R Schorlemmer, Nicholas Synovic, and George K Thiruvathukal. 2023. Reusing deep learning models: Challenges and directions in software engineering. In *2023 IEEE John Vincent Atanasoff International Symposium on Modern Computing (JVA)*. IEEE, 17–30.
- [32] Moming Duan, Qinbin Li, and Bingsheng He. 2024. Modelgo: A practical tool for machine learning license analysis. In *Proceedings of the ACM Web Conference 2024*. 1158–1169.
- [33] Margaret Mitchell Ezi Ozoani, Marissa Gerchick. 2022. Model Cards. <https://huggingface.co/blog/model-cards>.
- [34] Hugging Face. [n. d.]. User Studies. <https://huggingface.co/docs/hub/en/model-cards-user-studies>.
- [35] Hugging Face. 2024. Licenses. <https://huggingface.co/docs/hub/en/repositories-licenses>.
- [36] Hugging Face. 2024. Model Card Template. [https://github.com/huggingface/huggingface\\_hub/blob/main/src/huggingface\\_hub/templates/modelcard\\_template.md](https://github.com/huggingface/huggingface_hub/blob/main/src/huggingface_hub/templates/modelcard_template.md).
- [37] Hugging Face. 2025. Libraries. <https://huggingface.co/docs/hub/models-libraries>.
- [38] Hugging Face. 2025. Model sharing and uploading. [https://huggingface.co/transformers/v4.10.1/model\\_sharing.html](https://huggingface.co/transformers/v4.10.1/model_sharing.html).
- [39] Hugging Face. 2025. Pipelines. [https://huggingface.co/docs/transformers/main\\_classes/pipelines](https://huggingface.co/docs/transformers/main_classes/pipelines).
- [40] Hugging Face. 2025. Uploading models. <https://huggingface.co/docs/hub/en/models-uploading>.
- [41] Hugging Face. [n.d.]. Model Card Guidebook. <https://huggingface.co/docs/hub/en/model-card-guidebook>.
- [42] Hugging Face. [n.d.]. Model Cards. <https://huggingface.co/docs/hub/en/model-cards>.
- [43] Hugging Face. [n.d.]. Model Cards Writing Tool. [https://huggingface.co/spaces/huggingface/Model\\_Cards\\_Writing\\_Tool](https://huggingface.co/spaces/huggingface/Model_Cards_Writing_Tool).
- [44] Carlos Muñoz Ferrandis. [n. d.]. OpenRAIL: Towards open and responsible AI licensing frameworks. <https://opensource.org/ai/webinars/should-we-use-open-source-licenses-for-ml-ai-models>.
- [45] PyTorch Foundation. 2025. The Open Source Legacy and AI’s Licensing Challenge. <https://pytorch.org/blog/the-open-source-legacy-and-ais-licensing-challenge/>.
- [46] Kai Gao, Runzhi He, Bing Xie, and Minghui Zhou. 2024. Characterizing Deep Learning Package Supply Chains in PyPI: Domains, Clusters, and Disengagement. *ACM Trans. Softw. Eng. Methodol.* 33, 4 (2024), 97:1–97:27.
- [47] James G. Gatto. 2025. AI Drives Need for New Open Source Licenses – Linux Publishes the OpenMDW License. <https://natlawreview.com/article/ai-drives-need-new-open-source-licenses-linux-publishes-openmdw-license>.
- [48] Amir Gholami, Sehoon Kim, Zhen Dong, Zhewei Yao, Michael W Mahoney, and Kurt Keutzer. 2022. A survey of quantization methods for efficient neural network inference. In *Low-Power Computer Vision*. Chapman and Hall/CRC, 291–326.

- [49] glenn. 2004. Announcing (and explaining) our new 2.0 licenses. <https://creativecommons.org/2004/05/25/announcingandexplainingournew20licenses/>.
- [50] Micah Goldblum, Dimitris Tsipras, Chulin Xie, Xinyun Chen, Avi Schwarzschild, Dawn Xiaodong Song, Aleksander Madry, Bo Li, and Tom Goldstein. 2020. Dataset Security for Machine Learning: Data Poisoning, Backdoor Attacks, and Defenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45 (2020), 1563–1580. <https://api.semanticscholar.org/CorpusID:229934464>
- [51] Google Inc. 2024. TensorFlow Hub <https://www.tensorflow.org/hub>.
- [52] Grant Gross. 2007. Open-source legal group strikes again on BusyBox, suing Verizon. <https://www.computerworld.com/article/2537947/open-source-legal-group-strikes-again-on-busybox--suing-verizon.html>. Accessed: 2023-14-09.
- [53] Neil Gunningham, Robert A Kagan, and Dorothy Thornton. 2004. Social License and Environmental Protection: Why Businesses Go Beyond Compliance. *Law & Social Inquiry* 29, 2 (2004), 307–341. <https://doi.org/10.1111/j.1747-4469.2004.tb00338.x>.
- [54] Mary Hardy. [n. d.]. Should we use open source licenses for ML/AI models? [https://huggingface.co/blog/open\\_rail](https://huggingface.co/blog/open_rail).
- [55] Ahmed E Hassan, Dayi Lin, Gopi Krishnan Rajbahadur, Keheliya Gallaba, Filipe Roseiro Cogo, Boyuan Chen, Haoxiang Zhang, Kishanthan Thangarajah, Gustavo Oliva, Jiahuei Lin, et al. 2024. Rethinking software engineering in the era of foundation models: A curated catalogue of challenges in the development of trustworthy fmware. In *Companion Proceedings of the 32nd ACM International Conference on the Foundations of Software Engineering*. 294–305.
- [56] Geoffrey Hinton. 2015. Distilling the Knowledge in a Neural Network. *arXiv preprint arXiv:1503.02531* (2015).
- [57] Hugging Face Inc. 2024. Hugging Face <https://huggingface.co>. <https://huggingface.co>
- [58] ISO/IEC 5962:2021 2021. *Information technology – SPDX V2.2.1*. Standard. International Organization for Standardization, Geneva, CH.
- [59] Wenxin Jiang, Vishnu Banna, Naveen Vivek, Abhinav Goel, Nicholas Synovic, George K Thiruvathukal, and James C Davis. 2024. Challenges and practices of deep learning model reengineering: A case study on computer vision. *Empirical Software Engineering* 29, 6 (2024), 142.
- [60] Wenxin Jiang, Chingwo Cheung, Mingyu Kim, Heesoo Kim, George K Thiruvathukal, and James C Davis. 2024. Naming Practices of Pre-Trained Models in Hugging Face. *arXiv preprint arXiv:2310.01642* (2024).
- [61] Wenxin Jiang, Chingwo Cheung, George K Thiruvathukal, and James C Davis. 2023. Exploring naming conventions (and defects) of pre-trained deep learning models in hugging face and other model hubs. *arXiv preprint arXiv:2310.01642* (2023).
- [62] Wenxin Jiang, Jason Jones, Jerin Yasmin, Nicholas Synovic, Rajeev Sashti, Sophie Chen, George K Thiruvathukal, Yuan Tian, and James C Davis. 2023. PeaTMOSS: Mining Pre-Trained Models in Open-Source Software. *arXiv preprint arXiv:2310.03620* (2023).
- [63] Wenxin Jiang, Nicholas Synovic, Matt Hyatt, Taylor R. Schorlemmer, Rohan Sethi, Yung-Hsiang Lu, George K. Thiruvathukal, and James C. Davis. 2023. An Empirical Study of Pre-Trained Model Reuse in the Hugging Face Deep Learning Model Registry. *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)* (2023), 2463–2475. <https://api.semanticscholar.org/CorpusID:257365660>
- [64] Wenxin Jiang, Nicholas Synovic, Purvish Jajal, Taylor R Schorlemmer, Arav Tewari, Bhavesh Pareek, George K Thiruvathukal, and James C Davis. 2023. PTMTorrent: A Dataset for Mining Open-source Pre-trained Model Packages. In 2023 IEEE/ACM 20th International Conference on Mining Software Repositories (MSR). 57–61.
- [65] Wenxin Jiang, Nicholas Synovic, Rohan Sethi, Aryan Indarapu, Matt Hyatt, Taylor R Schorlemmer, George K Thiruvathukal, and James C Davis. 2022. An empirical study of artifacts and security risks in the pre-trained model supply chain. In *Proceedings of the 2022 ACM Workshop on Software Supply Chain Offensive Research and Ecosystem Defenses*. 105–114.
- [66] Wenxin Jiang, Jerin Yasmin, Jason Jones, Nicholas Synovic, Jiashen Kuo, Nathaniel Bielanski, Yuan Tian, George K Thiruvathukal, and James C Davis. 2024. Peatmoss: A dataset and initial analysis of pre-trained models in open-source software. In *2024 IEEE/ACM 21st International Conference on Mining Software Repositories (MSR)*. IEEE, 431–443.
- [67] Jason Jones, Wenxin Jiang, Nicholas Synovic, George Thiruvathukal, and James Davis. 2024. What do we know about Hugging Face? A systematic literature review and quantitative validation of qualitative claims. In *Proceedings of the 18th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*. 13–24.
- [68] Georgia Kapitsaki, Athina Paphitou, and Achilleas Achilleos. 2022. Towards open source software licenses compatibility check. In *Proceedings of the 26th Pan-Hellenic Conference on Informatics*. 96–101.
- [69] Georgia M Kapitsaki, Maria Papoutsoglou, Daniel M German, and Letteris Angelis. 2020. What do developers talk about open source software licensing?. In *2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, 72–79.
- [70] Kate Knibbs. 2025. Meta Secretly Trained Its AI on a Notorious Piracy Database, Newly Unredacted Court Docs Reveal. <https://www.wired.com/story/new-documents-unredacted-meta-copyright-ai-lawsuit/>

- [71] Jing Yu Koh. 2024. Model Zoo: Discover open source deep learning code and pretrained models <https://modelzoo.co>.
- [72] Keita Kurita, Paul Michel, and Graham Neubig. 2020. Weight Poisoning Attacks on Pretrained Models. *ArXiv abs/2004.06660* (2020). <https://api.semanticscholar.org/CorpusID:215754328>
- [73] Andrew M. St. Laurent. 2004. *Understanding Open Source and Free Software Licensing*. O'Reilly, Sebastopol, California.
- [74] Katherine Lee, A Feder Cooper, and James Grimmelmann. 2023. Talkin' 'Bout AI Generation: Copyright and the Generative-AI Supply Chain. *arXiv preprint arXiv:2309.08133* (2023).
- [75] Katherine Lee, A Feder Cooper, James Grimmelmann, and Daphne Ippolito. 2023. AI and Law: The Next Generation. *Available at SSRN 4580739* (2023).
- [76] Tao Liu, Chengwei Liu, Tianwei Liu, He Wang, Gaofei Wu, Yang Liu, and Yuqing Zhang. 2024. Catch the Butterfly: Peeking into the Terms and Conflicts among SPDX Licenses. *arXiv preprint arXiv:2401.10636* (2024).
- [77] Shayne Longpre, Robert Mahari, Anthony Chen, Naana Obeng-Marnu, Damien Sileo, William Brannon, Niklas Muennighoff, Nathan Khazam, Jad Kabbara, Kartik Perisetla, et al. 2023. The data provenance initiative: A large scale audit of dataset licensing & attribution in ai. *arXiv preprint arXiv:2310.16787* (2023).
- [78] Qinghua Lu, Liming Zhu, Ximei Xu, Yue Liu, Zhenchang Xing, and Jon Whittle. 2024. A taxonomy of foundation model based systems through the lens of software architecture. In *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering-Software Engineering for AI*. 1–6.
- [79] Tshilidzi Marwala, Eleonore Fournier-Tombs, and Serge Stinckwich. 2023. The use of synthetic data to train ai models: Opportunities and risks for sustainable development. *arXiv preprint arXiv:2309.00652* (2023).
- [80] Saeed Masoudnia and Reza Ebrahimpour. 2014. Mixture of experts: a literature survey. *Artificial Intelligence Review* 42 (2014), 275–293.
- [81] Heather Meeker. 2017. *Open source for business: a practical guide to open source software licensing*. CreateSpace.
- [82] Meta. 2024. META LLAMA 3 COMMUNITY LICENSE AGREEMENT. <https://llama.meta.com/llama3/license/>.
- [83] Microsoft. [n.d.]. trocr-base-handwritten. <https://huggingface.co/microsoft/trocr-base-handwritten>.
- [84] Microsoft. [n.d.]. trocr-large-stage1. <https://huggingface.co/microsoft/trocr-large-stage1>.
- [85] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. 2019. Model cards for model reporting. In *Proceedings of the conference on fairness, accountability, and transparency*. 220–229.
- [86] Christopher Moran. 2021. Machine Learning, Ethics, and Open Source Licensing (Part II/II). <https://thegradient.pub/machine-learning-ethics-and-open-source-licensing-2/>.
- [87] networkx 2024. NetworkX - Network Analysis in Python. <https://networkx.org/>.
- [88] Chinenye Okafor, Taylor R Schorlemmer, Santiago Torres-Arias, and James C Davis. 2022. Sok: Analysis of software supply chain security by establishing secure design properties. In *Proceedings of the 2022 ACM Workshop on Software Supply Chain Offensive Research and Ecosystem Defenses*. 15–24.
- [89] OpenAI. [n. d.]. clip-vit-large-patch14-336. <https://huggingface.co/openai/clip-vit-large-patch14-336>.
- [90] openai-terms 2023. Legalities of self improving agents. <https://community.openai.com/t/legalities-of-self-improving-agents/560666>.
- [91] Ernesto Lang Oreamuno, Rohan Faiyaz Khan, Abdul Ali Bangash, Catherine Stinson, and Bram Adams. 2024. The state of documentation practices of third-party machine learning models and datasets. *IEEE Software* (2024).
- [92] David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. 2021. Carbon Emissions and Large Neural Network Training. <http://arxiv.org/abs/2104.10350>
- [93] Federica Pepe, Vittoria Nardone, Antonio Mastropaoletti, Gabriele Bavota, Gerardo Canfora, and Massimiliano Di Penta. 2024. How do Hugging Face Models Document Datasets, Bias, and Licenses? An Empirical Study. In *Proceedings of the 32nd IEEE/ACM International Conference on Program Comprehension*. 370–381.
- [94] requests 2024. requests. <https://pypi.org/project/requests/>.
- [95] spdx 2025. SPDX. <https://spdx.dev>/
- [96] Trevor Stalnaker, Nathan Wintersgill, Oscar Chaparro, Massimiliano Di Penta, Daniel M German, and Denys Poshyvanyk. 2024. BOMs Away! Inside the Minds of Stakeholders: A Comprehensive Study of Bills of Materials for Software Systems. In *Proceedings of the 46th IEEE/ACM International Conference on Software Engineering*. 1–13.
- [97] Trevor Stalnaker, Nathan Wintersgill, Oscar Chaparro, Laura A. Heymann, Massimiliano Di Penta, Daniel M German, and Denys Poshyvanyk. 2025. Online replication package. [https://archive.softwareheritage.org/browse/origin/directory/?origin\\_url=https://github.com/TStalnaker44/hugging\\_face\\_analysis\\_replication](https://archive.softwareheritage.org/browse/origin/directory/?origin_url=https://github.com/TStalnaker44/hugging_face_analysis_replication).
- [98] Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, et al. 2024. Learning to (learn at test time): Rnns with expressive hidden states. *arXiv preprint arXiv:2407.04620* (2024).
- [99] Xin Tan, Kai Gao, Minghui Zhou, and Li Zhang. 2022. An exploratory study of deep learning supply chain. In *Proceedings of the 44th International Conference on Software Engineering*. 86–98.

- [100] Mina Taraghi, Gianolli Dorcelus, Armstrong Foundjem, Florian Tambon, and Foutse Khomh. 2024. Deep learning model reuse in the huggingface community: Challenges, benefit and trends. *arXiv preprint arXiv:2401.13177* (2024).
- [101] Mistral AI team. 2023. Mixtral of experts. <https://mistral.ai/news/mixtral-of-experts/>.
- [102] The Linux Foundation. 2024. PyTorch Hub <https://pytorch.org/hub>.
- [103] Jason Tsay, Alan Braz, Martin Hirzel, Avraham Shinnar, and Todd W. Mummert. 2020. AIMMX: Artificial Intelligence Model Metadata Extractor. In *MSR '20: 17th International Conference on Mining Software Repositories, Seoul, Republic of Korea, 29-30 June, 2020*. ACM, 81–92.
- [104] Ashlee Vance. 2010. The Defenders of Free Software. <https://www.nytimes.com/2010/09/26/business/26ping.html>. Accessed: 2023-14-09.
- [105] Christopher Vendome. 2015. A large scale study of license usage on github. In *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, Vol. 2. IEEE, 772–774.
- [106] Christopher Vendome, Daniel M German, Massimiliano Di Penta, Gabriele Bavota, Mario Linares-Vásquez, and Denys Poshyvanyk. 2018. To distribute or not to distribute? why licensing bugs matter. In *Proceedings of the 40th International Conference on Software Engineering*. 268–279.
- [107] Nick Vidal. 2023. The most popular licenses for each language in 2023. <https://opensource.org/blog/the-most-popular-licenses-for-each-language-2023>.
- [108] James Vincent. 2022. The lawsuit that could rewrite the rules of AI copyright. <https://www.theverge.com/2022/11/8/23446821/microsoft-openai-github-copilot-class-action-lawsuit-ai-copyright-violation-training-data>.
- [109] Ying Wang, Shing-Chi Cheung, Hai Yu, and Zhiliang Zhu. 2025. Managing Software Supply Chains. (2025).
- [110] Zhi Wang, Chaoge Liu, and Xiang Cui. 2021. EvilModel: Hiding Malware Inside of Neural Network Models. *2021 IEEE Symposium on Computers and Communications (ISCC)* (2021), 1–7. <https://api.semanticscholar.org/CorpusID:236087474>
- [111] Matt White, Ibrahim Haddad, Cailean Osborne, Xiao-Yang Yanglet Liu, Ahmed Abdelmonsef, Sachin Varghese, and Arnaud Le Hors. 2024. The Model Openness Framework: Promoting Completeness and Openness for Reproducibility, Transparency, and Usability in Artificial Intelligence. *arXiv:2403.13784 [cs.LG]* <https://arxiv.org/abs/2403.13784>
- [112] Kyle Wiggers. 2025. ‘Open’ AI model licenses often carry concerning restrictions. <https://techcrunch.com/2025/03/14/open-ai-model-licenses-often-carry-concerning-restrictions/>.
- [113] Laurie Williams, Giacomo Benedetti, Sivana Hamer, Ranindya Paramitha, Imranur Rahman, Mahzabin Tamanna, Greg Tystahl, Nusrat Zahan, Patrick Morrison, Yasemin Acar, et al. 2025. Research directions in software supply chain security. *ACM Transactions on Software Engineering and Methodology* 34, 5 (2025), 1–38.
- [114] Nathan Wintersgill, Trevor Stalnaker, Laura A Heymann, Oscar Chaparro, and Denys Poshyvanyk. 2024. “The Law Doesn’t Work Like a Computer”: Exploring Software Licensing Issues Faced by Legal Practitioners. *Proceedings of the ACM on Software Engineering* 1, FSE (2024), 882–905.
- [115] Yuhao Wu, Yuki Manabe, Tetsuya Kanda, Daniel M German, and Katsuro Inoue. 2017. Analysis of license inconsistency in large collections of open source projects. *Empirical Software Engineering* 22 (2017), 1194–1222.
- [116] Boming Xia, Tingting Bi, Zhenchang Xing, Qinghua Lu, and Liming Zhu. 2023. An empirical study on software bill of materials: Where we stand and the road ahead. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*. IEEE, 2630–2642.
- [117] Boming Xia, Dawen Zhang, Yue Liu, Qinghua Lu, Zhenchang Xing, and Liming Zhu. 2023. Trust in Software Supply Chains: Blockchain-Enabled SBOM and the AIBOM Future. *arXiv preprint arXiv:2307.02088* (2023).
- [118] Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. 2024. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities. *arXiv preprint arXiv:2408.07666* (2024).
- [119] Xinyu Yang, Weixin Liang, and James Zou. 2024. Navigating Dataset Documentations in AI: A Large-Scale Analysis of Dataset Cards on Hugging Face. *arXiv preprint arXiv:2401.13822* (2024).