Modelgo: A Tool for Machine Learning License Analysis

Moming Duan
National University of Singapore
Singapore
moming@nus.edu.sg

ABSTRACT

Productionizing machine learning projects is inherently complex, involving a multitude of interconnected components that are assembled like LEGO blocks and evolve throughout development lifecycle. These components encompass software, databases, and models, each subject to various licenses governing their reuse and redistribution. However, existing license analysis approaches for Open Source Software (OSS) are not well-suited for this context. For instance, some projects are licensed without explicitly granting sublicensing rights, or the granted rights can be revoked, potentially exposing their derivatives to legal risks. Indeed, the analysis of licenses in machine learning projects grows significantly more intricate as it involves interactions among diverse types of licenses and licensed materials. To the best of our knowledge, no prior research has delved into the exploration of license conflicts within this domain. In this paper, we introduce Modelgo, a practical tool for auditing potential legal risks in machine learning projects to enhance compliance and fairness. With Modelgo, we present license assessment reports based on 5 use cases with diverse model-reusing scenarios, rendered by XXX popular machine learning components. Finally, we summarize the reasons behind license conflicts and provide guidelines for minimizing them.

CCS CONCEPTS

• **Do Not Use This Code → Generate the Correct Terms for Your Paper**; *Generate the Correct Terms for Your Paper*; Generate the Correct Terms for Your Paper; Generate the Correct Terms for Your Paper.

KEYWORDS

Software licenses, Software reuse, Open source software, Model mining

1 INTRODUCTION

Over the past decade, the advancement and productization of AI infrastructures have significantly accelerated the proliferation of machine learning (ML) components [13], including AI models [21, 26], software [10, 27], and big datasets [7, 24]. Concurrently, the reuse of these components has gained popularity, motivated by concerns about their significant demands on financial and energy resources [25], as well as the widespread recognition of the value advocated by the open-source movement [22]. Unlike code reuse in the OSS field, the reuse of AI models follow a distinct schema. A frequently employed approach for AI models reuse is fine-tuning Pre-Trained Models (PTMs) [9, 26], where PTMs are adapted on a domain-specific dataset, leveraging their robust generalization capabilities.

From a legal perspective, model reuse is generally uncontroversial when its developers or affiliated companies own the copyright for all components. However, data and models often have separate copyright holders in nowadays ML projects [19, 20, 23, 29]. For instance, GPT-2 [19], developed by OpenAI, was trained on 45 million web pages containing content from third-party platforms like WordPress, GitHub, and IMDb, none of which is owned by OpenAI. These crowdsourced content typically provides limited usage and distribution rights to users through pre-agreed licenses (e.g., Creative Commons Licenses[1]), which may restrict certain reuse methods like remixing, reproducing, and translating. To prevent legal risk, it is essential to ensure that the final ML projects remain compaliant with all license conditions associated with the reused components [5, 14, 16].

However, compared to assessing licensing compliance for OSS, ensuring license compliance in ML projects poses several unique challenges. First, a ML project is not only a combination of software like an OSS project but also composed of datasets and models [9], which may be under different types of licenses (e.g., Free Content Licenses and AI model licenses [4]). Second, ML components often follow more complicated coupling paradigms and nested workflows. For instance, Openjourney[2] is an image generation model derived from StableDiffusion [21], and fine-tuned on images generated by another commercial product, Midjourney[3]. This demonstrates that knowledge can be transferred between models without explicit code integration [28]. Another challenge is improper and ambiguity licensing in ML projects. For example, GPT-2 and BERT [6] are regarded as part of software and then licensed as OSS (e.g., MIT and Apache-2.0). However, ML projects like StableDiffusion and Llama2 [26] tend to apply responsible AI restriction terms for both model and code, using AI model licenses such as OpenRAIL-M [4] and Llama2 Community License[4]. Additionally, to circumvent the limitations of standard OSS licenses, some licensors adopt non-commercial content licenses or custom licenses to protect the Intellectual Property (IP) of their models by prohibiting commercial use [11], fine-tuning [15], and reverse engineering [8]. Such ambiguity and the diverse licensing practices within ML projects increase significant legal uncertainty in license compliance analysis. As a result, traditional OSS license analysis approaches [16, 17] only consider inclusion and linking relationships among software and lack support for AI model licenses, making them unsuitable for ML project license analysis.

In this paper, we introduce Modelgo, a tool designed to analyze potential license conflicts, improper license choices, use restrictions and obligations in ML projects that involve nested component reuse procedures. To demonstrate the usefulness of Modelgo, we present

[1] https://creativecommons.org/licenses/ [2] https://openjourney.art/
[3] https://www.midjourney.com/ [4] https://huggingface.co/meta-llama/Llama-2-7b

5 use cases constructed using 15 datasets and 11 models from real-world scenarios, whose license types cover OSS, free content, and AI model. Our findings show that there exist potential legal risks when reusing components under copyleft or non-commercial licenses, and point out the need for attention to AI model licenses. The main contributions of our paper are:

- We raise the challenge of license analysis for ML projects and propose Modelgo to assessing it. To the best of our knowledge, our work is the first attempt to deal with this challenge in the ML context.
- As part of our work, we introduce a new taxonomy based on the forms of reused components to identify the corresponding conditions for various ML reuse mechanisms. This method helps mitigate ambiguity in cases of mismatch between applied license type and actual component type, allowing Modelgo to analyze components under various license types, including OSS, free content and AI models.
- We provide legal compliance assessment reports based on 5 use cases to showcase the effectiveness of our approach. Through our use cases, we offer valuable insights and experiences in achieving legal compliance in ML projects. Additionally, we also provide license choosing recommendations to minize the risk of non-compliance.

The rest of the paper is organized as follows. (TBD)

There is no consensus on whether the use of copyright works as input to train an AI system is an exercise of an exclusive right. There remains significant legal uncertainty about whether copyright applies to AI training, which means it may not always be clear whether a CC license applies. The larger model was trained on 256 cloud TPU v3 cores. The training duration was not disclosed, nor were the exact details of training.

Open source software license compliance [17]

The open source definition [18]

AFL [22]

Wudao2.0 1.75T MoE [FASTMOE: A FAST MIXTURE-OF-EXPERT TRAINING SYSTEM] [GLM-130B: AN OPEN BILINGUAL PRE-TRAINED MODEL]

Objectives and challenges associated with analyzing dataset license compliance? Getty Images (US), Inc. v. Stability AI, Inc. (1:23-cv-00135) Andersen et al v. Stability AI Ltd. et al (3:23-cv-00201) We are not aware of any copyright restrictions of the material

C4, Pile Common Crawl crowdsourced

COCO (CC-BY 4.0), CIFAR10 -> Flickr Unsplash License *Custom*: Compiling photos from Unsplash to replicate a similar or competing service. https://unsplash.com/license Pixabay License: Data mining, extraction, scraping and the use of programs or robots for automatic data collection and/or extraction of digital data on the Services and/or the content available therein is strictly prohibited for all purposes, including without limitation for machine learning purposes.

Google Street View (SVHN) https://about.google/brand-resource-center/products-and-services/geo-guidelines/

Software reuse is very simple from the legal point of view, if a company or an individual reuses software for which it has copyrights. However, things change dramatically if one wants to reuse software made by others, since software is protected by copyright and possibly by patents. Without explicit permission, no person other than the copyright holder is allowed to copy, distribute, or make derivative works from the original work.

2 BACKGROUND AND RELATED WORK

2.1 Machine Learning Project Licensing

Benjamoin *et al.* [2] propose Montreal Data License (MDL).

2.2 FOSS License Assessment

SPDX Automating the license compatibility process in open source software with SPDX

2.3 Machine Learning IP Protection

3 METHOD

This section is organized around three key questions in the context of ML license analysis: (i) How to determine the corresponding conditions in licenses for certain model reuse mechanisms? (ii) How to capture the dependency structure of a machine learning project? (iii) What types of non-compliance exist in ML projects and how to assess them? We will present our solutions to these questions in the following sections.

3.1 Taxonomy for ML License Analysis

Determining the corresponding conditions in licenses is a challenging task for ML projects due to the conceptual ambiguities in existing licensing language and the disorganization in current ML licensing practices. For example, CC-BY-ND prohibits the sharing of derivatives of licensed materials. However, its definition of making derivatives is unclear in the context of ML domain. For instance, should embeddings of a corpus be considered a derivative work upon that corpus? Unfortunately, even though Creative Commons provides a flow chart to illustrate the trigger conditions of CC licenses in the context of AI activity [3], it raises another question: *Is the output considered protectable copyright subject matter?* The answer depends on how the embedding activity is interpreted, for example, considering it as a translation of the original work can trigger the CC license.

MDL advocates the use of a *Top Sheet* to delineate what ML activities are allowed with data [2], but this proposal is rarely implemented in practice (things would be easier if it were widely accepted). Making things more complex, some projects release their models under free content licenses, like LayoutLMv3 model [11], which is licensed under CC-BY-NC-SA-4.0. This disorganization makes it unclear what kinds of ML activities can trigger licenses conditions in different contexts. An ideal and elegant solution would be to encourage licensors to make context-appropriate adaptations in their license agreements or terms of use to clarify the granted rights related to ML activities. However, some ML components may be composed of prior works that are shared under copyleft license templates, which may disallow such relicensing of their derivatives to a new license. Therefore, it is necessary to establish practical rules to bridge AI activities and existing licensing language.

To address the above challenge, we propose a result-based taxonomy that categorizes all AI activities into four categories based

Table 1: Summary of machine learning projects in Huggingface.

ML Project	Task	Data License	Software License	Model License	Dataset	Risk Resource
Stable Diffusion v1-5	Text to Image	CC-BY-4.0	CreativeML-OpenRAIL-M	CreativeML-OpenRAIL-M	LAION-5B	Common Crawl
BLOOM	Text Generation	*Mixture*	*Unknown*	BigScience-BLOOM-RAIL-1.0	*Crowdsourced*	Common Crawl, Wikipedia, etc.
OrangeMixs	Text to Image	*Mixture*	*Unknown*	CreativeML-OpenRAIL-M	*Crowdsourced*	Danbooru
ControlNet	Text to Image	*Unknown*	Apache-2.0	OpenRAIL	*Unknown*	n/a
Openjourney	Text to Image	CC-BY-NC-4.0	*Unknown*	CreativeML-OpenRAIL-M	Midjourney Gen	Midjourney Gen
ChatGLM-6B	Text Generation	*Mixture*	Apache-2.0	*Custom*	the Pile, Wudao, *Crowdsourced*	PubMed, Wikipedia, arXiv, GitHub, etc.
Llama2	Text Generation	*Unknown*	Llama2 Community License	Llama2 Community License	*Unknown*	n/a
StarCoder	Text Generation	*Mixture*	Apache-2.0	BigCode-OpenRAIL-M	The Stack	none
Falcon-40B	Text Generation	ODC-By	Apache-2.0	Apache-2.0	RefinedWeb	Wikipedia, Reddit, StackOverflow, etc.
Waifu Diffusion	Text to Image	*Mixture*	*Unknown*	CreativeML-OpenRAIL-M	*Unknown*	n/a
Dolly-v2-12B	Text Generation	CC-BY-SA-3.0&4.0	MIT	MIT	databricks-dolly -15k, the Pile	PubMed, Wikipedia, arXiv, GitHub, etc.
Dreamlike Photoreal	Text to Image	*Unknown*	*Unknown*	*Modified* CreativeML-OpenRAIL-M	*Unknown*	n/a
Counterfeit	Text to Image	*Unknow*	*Unknown*	CreativeML-OpenRAIL-M	*Unknown*	n/a
GPT-2	Text Generation	*Mixture*	*Modified* MIT	*Modified* MIT	*Crowdsourced*	WordPress, GitHub, wikiHow, IMDb, etc.
GPT-J-6B	Text Generation	*Mixture*	Apache-2.0	Apache-2.0	the Pile	PubMed, Wikipedia, arXiv, GitHub, etc.
LLaMA-7B	Text Generation	*Mixture*	*Custom*	*Custom*	*Crowdsourced*	GitHub, arXiv, etc.
BERT	Fill Mask	*Mixture*	Apache-2.0	Apache-2.0	Book Corpus, Wikipedia (en)	Wikipedia (en)
Whisper	ASR	*Unknown*	MIT	MIT	*Unknown*	n/a
MPT	Text Generation	*Mixture*	Apache-2.0	Apache-2.0	*Crowdsourced*	Common Crawl, Wikipedia, etc.

on the forms of their results. In our taxonomy, there are four categories of AI activities: Combination, Amalgamation, Distillation, and Generation, which are defined by four forms of their results, respectively: 1) Combination with strong separation; 2) Combination with weak separation; 3) Derivatives from concepts; and 4) Derivatives from data. Correspondingly, we can also categorize the usage behaviors in licensing language into these four categories based on their outcome forms.

We leverage Figure 1 to illustrate this idea. The left side consists of a list of AI activities, many of which pertain to model reusing methods, categorized based on the forms of their results. The middle part is our taxonomy that can classify these AI activities. Following this rule, we can also identify the corresponding terms in natural language license text shown on the right side. For example, Mixture of Experts (MoE) leverages a gating network to ensemble a batch of weak learners [12], which leads to a combination with strong separation and aligns with licensing terms like link, portion, collection, etc. Unlike combination, the results of amalgamation are difficult (or impossible) to separate, corresponding to AI activities such as modification, fine-tuning, model fusion, etc [5]. These unrecoverable revision of original works are corresponding license text like adapt, alter, remix, etc. Distillation and generation are derivatives of original works, which means the results will not contain any portion of the original works. These two AI activities are mostly defined in AI model licenses but are not covered by traditional OSS licenses and free content licenses.

By now, we can ascertain the suitable permissions, limitations, and responsibilities for each AI activity based on the language of the license, even when the license type isnt an exact match. However, its necessarily to emphasize three points. First, our proposed method only applies in cases where ambiguities exist in the definition. If the conditions of certain AI activities are explicitly defined in the license, then we should directly follow that. Second, due to the various definitions adopted in different licenses, the bridging rules depend on each specific case and may differ from Figure 1. Lastly, one AI activity may trigger multiple license conditions. For example, a fine-tuned model can be seen as a combination with weak separation of the original model, while it can also be viewed as a derivative from fine-tuning data. Therefore, we should design a mechanism to trace these multi-source dependency structures in ML projects, which we will detail in the next section.

3.2 Structure of ML Projects

ML projects have unique dependency relationships compared to OSS projects, like the dependencies between generated content and generation model, as well as between training data and trained model. We can summarize these dependencies in ML projects into three categories:

- **Mix-works** be embeded in the new work, either verbatim or in part, in a tangible form. They usually result from direct copying of original components or reusing them through AI activities like combination and amalgamation. These components are embedded into ML projects and must be released with the new work. For example, if we release a new work utilizing Mixture of Experts (MoE), it is equivalent to releasing all weak learners.
- **Sub-works** are similar to mix-works, but the difference is that they are not embedded in the new work. For instance, if we manage to release MoE model along with the data

[5] Whether embeddings constitute a combination with weak separation depends on the specific case. In Modelgo, we classify embeddings as amalgamation if they are created under a content license that treats translation as a form of modification.

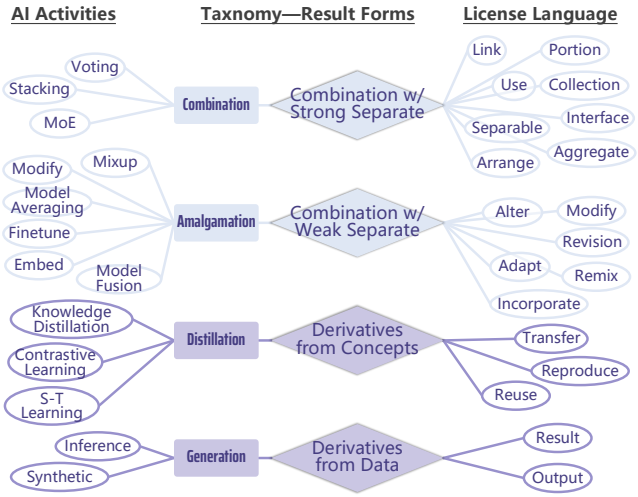

Figure 1: Our proposed taxonomy bridging AI activities and license terms based on their result forms.

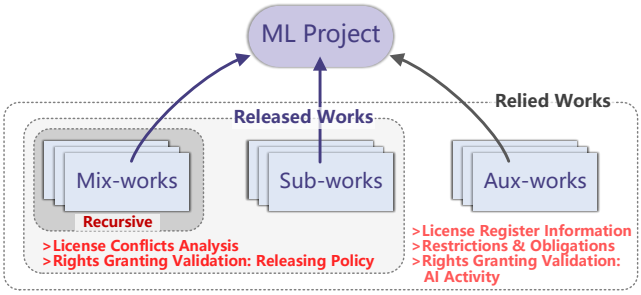

Figure 2: The proposed structure for capturing work dependencies in the context of ML projects with multiple reused components.

used for training the gating network, then this data will be regarded as the sub-works of MoE model.

- **Aux-works** are components used to build the new work and are either included in it or released with it. For example, the original model used for knowledge distillation.

Figure 2 illustrates the structure of a work constructed by reusing multiple components in the context of ML projects. The final ML project may be constructed through iterative reuse of other works, resulting in a ternary dependencies tree for this project. The reason we need this specially-designed tree structure is that works with different dependency types have different license condition proliferation rules, which need to be handled separately during subsequent license analysis.

3.3 License Analysis in ML Projects

We have outlined all the necessary preparation steps for license analysis in previous sections. Their detailed implementations in Modelgo are as follows.

Preparation Step 1: Following our proposed taxonomy, we manually transcribed the terms in the license text to a standard machine-readable file in YAML format [6]. This file contain following informations for each license:

- Basic license descriptions, including its name, SPDX short ID, license version, license types (e.g., public domain, permissive, copyleft, proprietary), preferred work types (e.g., software, data, model), and supporting labels such as *disclose code required* or *auto-relicensing applied*.
- Rights granting information, including granted rights and reserved rights as defined by the license, along with the permitted reusing methods or result forms for redistribution. The prefix of such granting also be noted for cases where the granted rights can be revoked.
- Corresponding terms for each AI activity, which contain result forms and relicensability of the activity, corresponding restrictions, and obligations. This item will be marked as *No Defined* if both the activity and the result forms of this activity are not explicitly covered in the license text.

Preparation Step 2: To capture the dependency structure of works as shown in Figure 2, we encode the rules of dependencies construction for each AI activity. For example, if we generate embeddings of a corpus using an NN model, then the corpus is considered the sub-work of the generated embeddings, with the activity labeled as *embed*, and the NN model is categorized as the aux-work with the activity labeled as *use*. Furthermore, if the corpus is a collection of smaller corpora, then these smaller corpuses are categorized as the mix-works of the integrated corpus, with the activity labeled as *combine*. By recursively traversing this dependencies tree, we can gather all the dependent works and the activities used to build this ML project.

It is important to emphasize a concept in our license analysis approach called *activity proliferation*, which means that the activity performed by a work will recursively proliferate to all its mix-works. In the example of the corpus collection mentioned above, the *embed* calculation performed on the collection will be applied to all the smaller corpuses, triggering their license conditions related to *embed* as well. Similarly, as shown in Figure 2, all rights granting validation and license conflicts analysis of a work should be proliferated to all its mix-works. On the other hand, aux-works are not released with the project, so they are out of the scope of license conflict analysis and rights granting validation for release. In summary, mix-works, sub-works, and aux-works have different scopes in ML license analysis, which is why we need to distinguish between them.

Analysis Step: Given the license information and dependencies tree of ML projects, we are ready to analyze the license conflicts within it. Modelgo's license analysis consists of three phases:

Initial phase, where we register each component with clear license name, version, type, and format, and then construct their dependencies using our predefined reusing functions. The release policy should be preset here, and we support personal use, sharing, and selling. Normally, few conditions apply when you only use the work personally, and most licenses limit behaviors like redistribution and commercial use.

[6] We attempted to use chatGPT to generate this content, but it often behaved unreliably in understanding our taxonomy and produced some stochastic answers [1].

Table 2: License warnings, errors, restrictions/obligations, and notices assessed by Modelgo in initial phase, license determination phase and license validation phase.

Warning, Error, Restriction, Notice	Description
Copyleft / Revocable / No Public Notice	This license or its granted rights are **copyleft / revocable / no public**.
License Type Mismatch Warning	License preferred work type is **not compatible** with this work type.
License Disclose Self Warning	License requires this work (in binary or SaaS format) to remain **open source** or provide a **readable copy** of the source code.
Rights Not Granted Warning	License of this work does **not explicitly grant** you the right to do (...)
Rights Not Granted Error	License of this work **cannot grant** you the right to do (...)
Multiple Copyleft Licenses Error	Work has a license conflict as it involves **multiple copyleft** licenses.
Cannot Share Error	License **prohibits sharing** of this work.
State Changes Restriction	This work must **state changes** according to related license(s).
Include License Restriction	This work must retain the **original license file** according to the related license(s).
Include Notice Restriction	This work must retain all **notice files** (may contain copyright, patent, trademark and attribution) according to the related license(s).
Use Behavioral Restriction	This work must comply with the **use restriction** terms according to related license(s).
Runtime Restriction	This work must comply with the **rumtime restriction** terms according to related license(s).

Table 3: List of licenses supported by Modelgo, covering over 98% of licensed models and datasets on Huggingface.

OSS License (99.8%)	**Content License** (99.2%)	**AI Model License** (98.2%)
Apache-2.0, Unlicense, MIT, AFL-3.0, GPL-3.0, AGPL-3.0, LGPL-3.0, LGPL-2.1, BSD-3-Clause, BSD-3-Clause-Clear, BSD-2-Clause, Artistic-2.0, WTFPL-2.0, OSL-3.0, ECL-2.0	CC0-1.0, CC-BY-4.0, CC-BY-SA-4.0, CC-BY-NC-4.0, CC-BY-ND-4.0, CC-BY-NC-ND-4.0, CC-BY-NC-SA-4.0, PDDL, C-UDA, LGPL-LR, GFDL	OpenRAIL++, CreativeML-OpenRAIL-M, BigScience-BLOOM-RAIL-1.0, Llama2, OPT-175B, SEER

License determination phase, where we iteratively derive the appropriate new licenses for intermediate reused results. Copyleft proliferation occurs when there is a triggered copyleft license in the relied components. An error will raise if there are other copyleft licenses or if there are components that cannot be relicensed. To condense our analysis results, we prioritize using *Unlicense* for intermediate results once they are relicenseable. After this phase, all components and their derivatives should have a well-determined license name.

License validation phase, where we validate the required rights for construct and release this project whether can be granted. The validation also includes compliance with disclosure requirements, such as when a components is in binary format but subject to conditions that require source code disclosure. The releaseability of the final result will be validated upon its mix-works and sub-works, and then an assessment report will be generated.

Table 2 presents the warnings, errors, restrictions, obligations, and notices that can be detected using Modelgo. Table 3 lists the licenses supported by Modelgo, which collectively cover over 98% of licensed models and datasets on Huggingface [7]. In the next section, we will present five case studies based on real ML components.

[7] Licenses without clear names and versions are excluded from the calculation. Worth mentioning, our coverage represents only 24.8% and 5.2% of the models and datasets on the entire site due to the significant number of works without license information.

Table 4: Specifications of AI components used in case studies, which include Copyleft License, Permissive License, Public Domain Licens and No public license.

Work Name	License Name	Type	Modality/Usage
Wikipedia	CC-BY-SA-4.0	Data	Text
StackExchange	CC-BY-SA-4.0		
FreeLaw	CC-BY-ND-4.0		
arXiv	CC-BY-NC-SA-4.0		
PubMed	CC-BY-NC-SA-4.0		
Deep-sequoia	CC-BY-NC-ND-4.0		
Midjourney Gen	CC-BY-NC-ND-4.0		Image
Flickr	CC-BY-NC-SA-4.0		
StockSnap	CC0-1.0		
Wikimedia	CC-BY-SA-4.0		
OpenClipart	CC0-1.0		
ccMixter	CC-BY-NC-4.0		Voice
Jamendo	CC-BY-NC-ND-4.0		
Thingverse	CC-BY-NC-SA-4.0		3D model
Vimeo	CC-BY-NC-ND-4.0		Video
Baize	GPL-3.0	Model	Text Generation
BLOOM	BigScience-BLOOM-RAIL-1.0		
Llama2	Llama2		
BigTranslate	GPL-3.0		
BERT	Apache-2.0		Fill-Mask
Stable Diffusion	CreativeML-OpenRAIL-M		Text to Image
MaskFormer	CC-BY-NC-4.0		Image Segmentation
DETR	Apache-2.0		
Whisper	MIT		Voice to Text
X-Clip	MIT		Video to Text
I2VGen-XL	CC-BY-NC-ND-4.0		Image to Video

4 CASE STUDY DETAILS

An ideal practice of Modelgo is to assess real-world ML projects and detect their potential license compliance issues. However, this can be challenging in practice due to three present situations:

(1) Prevalent Licensing Disorganization in ML Projects: Many ML projects lack organized licensing information, making it difficult to ascertain the licenses of individual components.

(2) Lack of Development Lifecycle Information for ML Reusing: ML reusing often occurs without a clear record, making it hard to trace the origins and licenses of components used.

(3) Non-compliance within Datasets: Crowdsourced datasets often suffer from license non-compliance issues [20], making the licenses (usually permissive) declared by dataset collectors invalid.

Consequently, directly analyzing real-world ML projects may result in uncertainty, over-optimistic results, and often fail to detect any license conflicts. Therefore, to validate Modelgo, we have designed five ML scenarios rendered using 15 common data sources and 11 models that cover 5 modalities and 7 tasks, respectively. Table 4 shows the specifications of the involved data sources and models, whose licenses include copyleft, permissive, public domain, and no public license [8]. Furthermore, our case studies can cover all events listed in Table 2, and the their details and findings are provided in the following section.

4.1 CASE I : Corpus Combination

Our

[8] Some data sources contain crowdsourced content with multiple licenses, and we selected a non-public domain license among them.

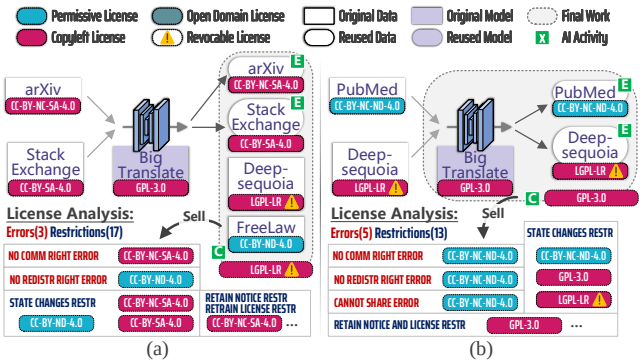

Figure 3: Case Study I: Corpus Combination. (a) LGPL-LR proliferation, CC collection; (b) LGPL-LR no linguistic resource, CC No redistribution.

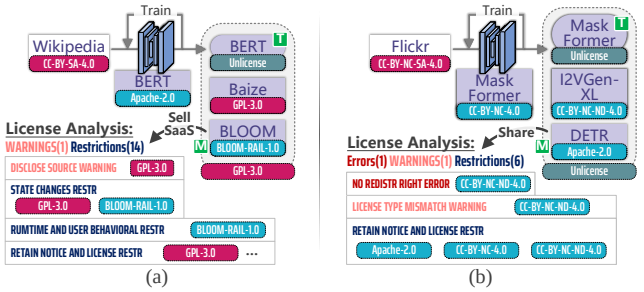

Figure 4: Case Study II: Mixture of Experts. (a) BLOOM-RAIL, binary of GPL; (b) Unlicense, CC-BY-NC no distribute derivative. GPL Automatic Licensing of Downstream Recipients

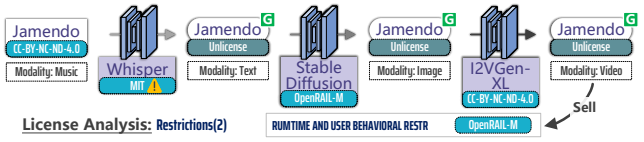

Figure 5: Case Study III: Pipeline.

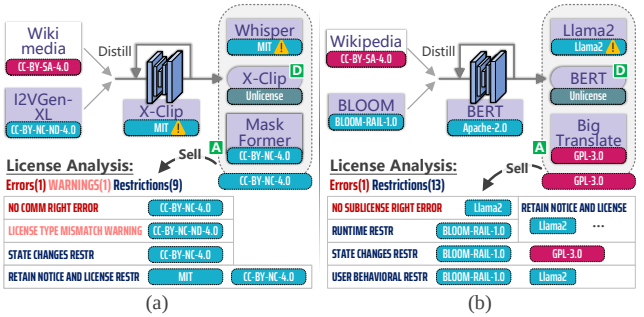

Figure 6: Case Study IV: distillation and model averaging.

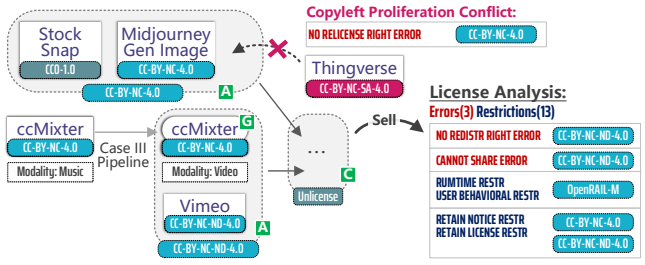

Figure 7: Case Study V: distillation and model averaging.

5 DISCLAIMER

The content presented in this article is intended for general informational purposes only and should not be construed as legal advice. Any views, opinions, findings, conclusions, or recommendations expressed in this material are the sole responsibility of the author(s) and do not represent the perspectives of any organization or entity.

REFERENCES

[1] Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM conference on Fairness, Accountability, and Transparency (FAccT)*. 610–623. https://doi.org/10.1145/3442188.3445922

[2] Misha Benjamin, Paul Gagnon, Negar Rostamzadeh, Chris Pal, Yoshua Bengio, and Alex Shee. 2019. Towards standardization of data licenses: The montreal data license. *arXiv preprint arXiv:1903.12262* (2019).

[3] Creative Commons. 2023. Artificial intelligence and CC licenses. Retrieved September 25, 2023 from https://creativecommons.org/faq/#artificial-intelligence-and-cc-licenses

[4] Danish Contractor, Daniel McDuff, Julia Katherine Haines, Jenny Lee, Christopher Hines, Brent Hecht, Nicholas Vincent, and Hanlin Li. 2022. Behavioral use licensing for responsible AI. In *2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*. 778–788. https://doi.org/10.1145/3531146.3533143

[5] Xing Cui, Jingzheng Wu, Yanjun Wu, Xu Wang, Tianyue Luo, Sheng Qu, Xiang Ling, and Mutian Yang. 2023. An Empirical Study of License Conflict in Free and Open Source Software. In *2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*. IEEE, 495–505. https://doi.org/10.1109/ICSE-SEIP58684.2023.00050

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 17th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. 4171–4186. https://doi.org/10.18653/v1/n19-1423

[7] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2020. The Pile: An 800GB Dataset of Diverse Text for Language Modeling. *arXiv preprint arXiv:2101.00027* (2020).

[8] Priya Goyal, Quentin Duval, Isaac Seessel, Mathilde Caron, Mannat Singh, Ishan Misra, Levent Sagun, Armand Joulin, and Piotr Bojanowski. 2022. Vision models are more robust and fair when pretrained on uncurated images without supervision. *arXiv preprint arXiv:2202.08360* (2022).

[9] Xu Han, Zhengyan Zhang, Ning Ding, Yuxian Gu, Xiao Liu, Yuqi Huo, Jiezhong Qiu, Yuan Yao, Ao Zhang, Liang Zhang, et al. 2021. Pre-trained models: Past, present and future. *AI Open* 2 (2021), 225–250. https://doi.org/10.1016/j.aiopen.2021.08.002

[10] Jiaao He, Jidong Zhai, Tiago Antunes, Haojie Wang, Fuwen Luo, Shangfeng Shi, and Qin Li. 2022. FasterMoE: modeling and optimizing training of large-scale dynamic pre-trained models. In *Proceedings of the 27th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP)*. 120–134. https://doi.org/10.1145/3503221.3508418

[11] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. 2022. LayoutLMv3: Pre-training for document ai with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia (MM)*. 4083–4091. https://doi.org/10.1145/3503161.3548112

[12] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation* 3, 1 (1991), 79–87. https://doi.org/10.1162/neco.1991.3.1.79

[13] 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. In *Proceedings of the 45th IEEE/ACM International Conference on Software Engineering (ICSE)*. 2463–2475. https://doi.org/10.1109/ICSE48619.2023.00206

[14] Georgia M Kapitsaki, Frederik Kramer, and Nikolaos D Tselikas. 2017. Automating the license compatibility process in open source software with SPDX. *Journal of Systems and Software (JSS)* 131 (2017), 386–401. https://doi.org/10.1016/j.jss.2016.06.064

[15] Dreamlike Tech Ltd. 2023. Dreamlike Photoreal 2.0. Retrieved September 25, 2023 from https://huggingface.co/dreamlike-art/dreamlike-photoreal-2.0

[16] Arunesh Mathur, Harshal Choudhary, Priyank Vashist, William Thies, and Santhi Thilagam. 2012. An empirical study of license violations in open source projects. In *2012 35th Annual IEEE Software Engineering Workshop (SEW)*. IEEE, 168–176. https://doi.org/10.1109/SEW.2012.24

[17] Philippe Ombredanne. 2020. Free and open source software license compliance: tools for software composition analysis. *Computer* 53, 10 (2020), 105–109. https://doi.org/10.1109/MC.2020.3011082

[18] Bruce Perens. 1999. The open source definition. *Open sources: voices from the open source revolution* 1 (1999), 171–188.

[19] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.

[20] Gopi Krishnan Rajbahadur, Erika Tuck, Li Zi, Dayi Lin, Boyuan Chen, Zhen Ming, Daniel M German, et al. 2021. Can I use this publicly available dataset to build commercial AI software?–A Case Study on Publicly Available Image Datasets. *arXiv preprint arXiv:2111.02374* (2021).

[21] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10684–10695. https://doi.org/10.1109/CVPR52688.2022.01042

[22] Lawrence Rosen. 2005. *Open Source Licensing: Software Freedom and Intellectual Property Law*. Prentice Hall Professional Technical Reference, New Jersey.

[23] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. BLOOM: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100* (2022).

[24] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. 2022. LAION-5B: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems (NeurIPS)* 35 (2022), 25278–25294.

[25] Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and Policy Considerations for Deep Learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*. 3645–3650. https://doi.org/10.18653/v1/p19-1355

[26] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288* (2023).

[27] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 38–45. https://doi.org/10.18653/v1/2020.emnlp-demos.6

[28] Shan You, Chang Xu, Fei Wang, and Changshui Zhang. 2021. Workshop on Model Mining. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 4177–4178. https://doi.org/10.1145/3447548.3469471

[29] Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2023. GLM-130B: An Open Bilingual Pre-trained Model. *Proceedings of the 11th International Conference on Learning Representations (ICLR)*.