# The GitHub Copilot Case
# Going from Software Protection to Artificial Intelligence Authorship

Gabriele Montanari

## Table of Contents



## Summary

A recent class-action against GitHub Copilot has given new fuel to the discourse around AIs and copyright. In this case, Copilot is an AI that has been trained on a large number of publicly available source codes, including GitHub's public repositories. Since these codes are protected by open-source licenses, legal problems arise both from the AI training with licensed material in the input phase and from how Copilot outputs verbatim snippets of code deprived of the related Copyright Management Information.

Regarding input infringement, in the US, text and data mining is a matter of fair use. On this regard, *Authors Guild v. Google, Inc*. and *Google LLC v. Oracle America, Inc*. give a blueprint on how fair use jurisprudence relates to TDM and protected software. More in detail, software suffers from the "original sin" of being considered a functional product. Arguably, US jurisprudence undervalues the fact that what is copyrighted is not the function, but the specific expression of it. In fact, in *Oracle*, the majority decision did not reward how the declaring code of the Java API was organized in an intuitive and understandable way that made it so appreciated by developers. These judgements revolve around the orientation that copyright's ultimate goal is to expand public knowledge and understanding, and authors are not the ultimate beneficiaries of it. Applying analogous reasonings to the Copilot case, it feels safe to assume that an US court would consider TDM in this instance fair use. Even so, other instances of TDM might receive different evaluations.

On the other hand, in the EU, TDM has been the object of a specific provision, with Articles 3 and 4 of Directive 2019/790/EU. Article 4 provides an exception to the right to reproduction, of which everyone can be the beneficiary, but restricted to lawfully accessible works. At the same time, rightsholders can expressly reserve the use of their works, exercising their right to opt-out from mining activities. Accordingly, publishing code on a public GitHub repository, and consequently licensing the software to GitHub, can be also interpreted as allowing mining on the published code. But the matter is not so simple, since it often happens that programmers add to their repositories even code written by third parties. Additionally, the very adequacy of these TDM exceptions is still debated. It is feared that a less aggressive implementation of mining would translate in a loss of market opportunities for European countries. Furthermore, there is no definitive answer on whether AI training is included in the scope of these exceptions, even if AI companies will probably assume that it is. This discussion extends to the actual scope of protection of the right to reproduction, since an antithesis to the technical and literal reading of the right has found its way in the *Pelham* case.

On the output side, what is relevant for the US is the possible violation of sections 1201–1205 of title 17 U.S.C. as amended by the DMCA. In fact, it can be affirmed that GitHub/Microsoft is distributing a product that circumvents the license system that governs the open-source ecosystem. Again, this becomes a legal problem of whether the occasional reproduction of licensed content, deprived of its CMI, counts as fair use. According to *HathiTrust*, the creation of complete digital copies of protected works can be transformative when they serve "a new and different function from the original work". So, it can be argued that verbatim reproduction of software is not an "extraction of information"; it does not offer a different function and it should not be considered transformative. Moreover, the lack of the CMI is not functional for the services that the AI offers. This absence is a choice of the AI's producers, and not an essential part of machine learning. This shortcoming affects also the fourth factor of the fair use test, since defining what Copilot does as admissible would mean validating all similar tools, capable of bypassing licenses on software, that will multiply exponentially in the next years. Still, in order to find a DMCA violation, the substance of the reproduction must be considered. Even a small amount of copying can be considered outside the scope of fair use

when the copied fragment represents the "heart" of the original's authorial expression. Another interesting question on the output side is about authorship on AI output. Since the creative power of the human mind is considered indispensable for authorship, it must be concretely evaluated what is the level of human input and supervision on the code that is produced by Copilot. However, the machine cannot be fully equated to "a tool like a pen". When it reproduces licensed code without giving its users any kind of warning, the fault can be only of the machine's producers.

The European stance on authorship is similar to the American one, therefore programmers must use their own personal capabilities to define the final form of what is outputted by Copilot. While infringement of the right to reproduction on Copilot's side must be also assessed in concrete. The practical or utilitarian function of a work does not impede its copyright protection, except when the expression is dictated only by a technical function. This EU orientation brings to the conclusion that even reduced fragments of code can be eligible for copyright protection. Nevertheless, it must be gauged case-by-case how much the programmer added his personal touch to the code arrangement.

In conclusion, the Copilot case is one of the testing grounds for the policy decisions that are being made in these years. With the warning that, in this instance, the open-source ecosystem could be severely damaged. First off, there are still uncertainties on when code is expressive and when instead it is "inherently bound together with uncopyrightable ideas". And, secondly, there are conflicting views on whether protecting authorship is more important than catching up with other countries. Especially with countries like China making bold decisions like the *Dreamwriter* case, the US and EU must determine where they stand.

## 1. The GitHub Copilot AI, a tool to help programmers or a washing machine for copyrighted material?

On the 21st of June 2022, GitHub made its AI tool available to all programmers around the world, putting it on the market as a subscription-based service[1]. The official website describes

[1] https://github.blog/2022-06-21-github-copilot-is-generally-available-to-all-developers/. GithHub Copilot was originally announced for technical preview on the 29th of June 2021. D. GERSHGORN, "GitHub and OpenAI launch a new AI tool that generates its own code", *The Verge*, June 29 2021, https://www.theverge.com/2021/6/29/22555777/github-openai-ai-tool-autocomplete-code.

GitHub Copilot as an artificial intelligence "pair programmer that helps you write code faster and with less work"[2].

To go more into detail, Copilot is an AI that is powered by a modified version of an OpenAI's product called Codex. Both have been trained on a large number of publicly available source codes. By the company's own admission, GitHub's repositories are included in the training material[3].

It is important to understand that open-source code, even if freely accessible, is not a work in the public domain. On the contrary, an open-source license implicates specific criteria for distribution of protected software. This is a regime that facilitates the exchange of ideas and inputs between different programmers, but the authors' creative work still gets protected under different terms depending on the specific open-source licenses[4].

Even if conditions and limitations on use and distribution of software are variable, some notable ones are: the inclusion of a copy of the license and copyright notice, when distributing the protected material; releasing modifications to the code under the same license, when distributing it; documenting all changes made to the software[5].

The license, that a programmer chooses when he creates a new repository on GitHub, is still valid for the myriads of codes that have been used to train Copilot. So, it becomes apparent that the crux of the problem is how Copilot has been trained. Being fed licensed material, this AI will learn to reason on protected software. Furthermore, it can be argued that whatever comes out of Copilot consists in modified versions of licensed codes, deprived of the notices and licenses that should accompany them.

In light of these possible infringements, on the 3rd of November of 2022, a class-action lawsuit was filed against GitHub, Microsoft, and OpenAI. The claims for relief of this lawsuit comprehend: violation of Section 1202 of the Digital Millennium Copyright Act, and breach of contract for violation of the open-source licenses[6].

The purpose of this paper is not to assess the validity of these claims, or to back them up. But, instead, to assess the different legal problems coexisting in this scenario and to evaluate them under a comparative lens, taking in consideration both US and EU law. This work is structured in a first section where I will examine the possibility of copyright infringement when protected software is used as input material to train an AI, and whether fair use doctrine is applicable in this instance. In a second section, I will reason on the code that is outputted by Copilot, if it ends up being a verbatim copy of licensed code, because in this case both infringement and ownership questions would arise. In a final section, I will review my findings and assess the impact that certain policy decisions can have on the future.

---

[2] https://github.com/features/copilot. According to reports coming from programmers that worked with Copilot, the tool functions as an auto-compiler, suggesting the next lines of code when someone starts typing. But it also has other capabilities, like improving a given code, substituting the unnecessary parts with a single command. C. THOMPSON, "It's Like GPT-3 but for Code—Fun, Fast, and Full of Flaws", *Wired*, March 15 2022, https://www.wired.com/story/openai-copilot-autocomplete-for-code/.

[3] https://github.com/features/copilot.

[4] https://opensource.org/osd.

[5] https://choosealicense.com/licenses/.

[6] https://githubcopilotlitigation.com/.

## 2. Input infringement, can an AI learn from copyrighted material without infringing the rights of the authors?

### 2.1 Solution under US Law

The recent class-action rises no specific claims related to the fact that GitHub Copilot itself is "composed" by the protected material it has been trained with. In fact, the code that the users originally stored in their GitHub repositories has been used to establish the weights of the neural network. Probably, the plaintiffs did not focus on this particular issue because, even if there is no clear decision on whether copyrighted material can be used to train AIs, the United States jurisprudence seems inclined to admit text and data mining as fair use[7]. But let us proceed with order.

Software is a curious creature that lives in a grey zone between patents and copyright, since its creation mixes functional aspects with creative ones. It was during the 1970s that the discussion around this topic became particularly vivid, with experts divided on what could be the best tool to protect authors (or inventors) of computer programs between: copyright, patents, and a specific sui generis right. The US was one of the most pioneer countries, since it was in 1974 that the Congress created the National Commission on New Technological Uses of Copyrighted Works (CONTU). The CONTU issued its final report on this subject in 1978, and that paved the road to Congress amending the Copyright Act of 1976, giving a legal definition to "computer program". The rest of world soon followed suit, recognizing software copyright protection even in international treaties[8].

But these developments did not erase some uncertainties that eventually found their voice in case law. This is especially true for the case of *Google LLC v. Oracle America, Inc*. In that instance, the US Supreme Court concluded that Google copying 11,500 lines of code from the Java SE API (0,4% of the lines of the software interface) constituted fair use. The copied part of the declaratory code was needed so that programmers, familiar with Java, could use the same commands when working on the new Android platform. This interpretation of code as "functional to something" clears the way to other possible judgements that will allow fair use of code. As Justice Thomas highlighted in his dissenting opinion, the Supreme Court chose to bypass the question of whether an API is copyrightable or not; but it then overcorrected itself, penalizing the API excessively when assessing the nature of copyrighted work during the fair use judgement. In this sense we can see how, in US jurisprudence, copyrighted software suffers from the "original sin" of being a functional product. Arguably, not enough weight is

[7] P. SAMUELSON, "The EU's Controversial Digital Single Market Directive – Part II: Why the Proposed Mandatory Text- and Data-Mining Exception Is Too Restrictive", *Kluwer Copyright Blog*, July 12 2018, https://copyrightblog.kluweriplaw.com/2018/07/12/eus-controversial-digital-single-market-directive-part-ii-proposed-mandatory-text-data-mining-exception-restrictive/.

[8] H.Y. CHEN, *Copyright Protection for Software 2.0? Rethinking the Justification of Software Protection under Copyright Law*, in J.A. LEE (ed.), R. HILTY (ed.), K.C. LIU (ed.), *Artificial Intelligence and Intellectual Property*, Oxford, online edn, Oxford Academic, 2021, 325-328.

given to the fact that what is copyrighted is not the function, but the specific expression of it. In particular, this majority decision did not reward how the declaring code was organized in an intuitive and understandable way that made it so appreciated by developers.

As hinted above, the problematics of copyright and software broaden into matters of fair use. The US orientation is that "[t]he ultimate goal of copyright is to expand public knowledge and understanding" and "while authors are undoubtedly important intended beneficiaries of copyright, the ultimate, primary intended beneficiary is the public, whose access to knowledge copyright seeks to advance by providing rewards for authorship"[9]. The very purpose of fair use itself is to individuate instances where unauthorized copying should be permitted, in order "[t]o promote the Progress of Science and useful Arts"[10]. In that way, fair use is an essential element in arguments that put public interests ahead of those of the private author. Text and data mining (TDM) has been considered fair use also in *Authors Guild v. Google, Inc*. Even if that case regarded the construction of a database that permitted the public to expand its knowledge on literature for free.

Indeed, the question of whether using licensed software for AI training is illicit, consists in a question of whether TDM is fair use or not. Even so, GitHub's situation merits to be examined for its own merits[11]. Pursuant to 17 U.S.C. §107, fair use is an assessment that is conducted considering four different factors (the purpose and character of the use, the nature of the copyrighted work, the amount and substantiality of the portion used related to the copyrighted work as a whole, the effect of the use upon the potential market for or the value of the copyrighted work). In this evaluation, the first and the fourth factors are considered the most important[12].

Starting from the first factor, an instance of copying is called "transformative" when it adds something new and important to the original work[13]. It can be successfully argued that using existing code to train an AI, that will help programmers producing new code faster, is perfectly aligned with the purpose of copyright in the American system, giving new purpose and meaning to existing work. It can be intended as a use that fuels "that creative 'progress' that is the basic constitutional objective of copyright itself", bringing further growth to the market[14]. On the other hand, the fact that Copilot is offered as a service on a paid subscription and that GitHub's good faith is questionable could give a fair push in determining the absence of fair use. However, similar circumstances were dismissed in the comparable *Oracle* case,

[9] *Authors Guild v. Google, Inc.*, No. 13-4829 (2d Cir. 2015).

[10] *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S. 569, 575 (1994) (quoting U.S. Const., Art. I, § 8, cl. 8).

[11] Fair use doctrine itself encourages its appliance on a case-by-case basis, determining the contextual limits of the "copyright monopoly". This adaptability is even more important in a scenario of continuous technological development. *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S., at 577–578; H. R. Rep. No. 94–1476, 65–66 (1976).

[12] The fourth factor is "[…] the single most important element of fair use", according to *Harper & Row Publishers, Inc. v. Nation Enterprises*, 471 U.S. 539, 566 (1985) (citing MELVILLE B. NIMMER, *3 Nimmer on Copyright § 13.05[A]*, at 13–76 (1984)). While the relevance of the first factor is highlighted in *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S. at 591. Because the more the new use is transformative, the less it will appear as a substitute for the original work.

[13] What is weighted is whether the new work "adds something new, with a further purpose or different character, altering the first with new expression, meaning or message." *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S., at 579.

[14] *Google LLC v. Oracle America, Inc.*, 593 U.S., (2021), citing *Feist Publications, Inc. v. Rural Telephone Service Co.*, 499 U.S., 349-350.

where the majority underlined that many fair uses are commercial in nature[15] and that "[c]opyright is not a privilege reserved for the well-behaved"[16]. If that jurisprudence orientation is followed through, the transformative use of producing novel intelligence will prevail over other considerations.

Secondly, "the nature of the copyrighted work". In *Oracle*, the Sun Java API was considered an easy target for fair use because of its functional nature. Even if this choice was portrayed as an exception, the underlying reasoning still arises perplexities. The majority labelled the product as "inherently bound together with uncopyrightable ideas", but at the same time recognized that creative work went into making this code attractive to programmers. What makes this position so particular is that, arguably, most source codes are highly functional and "inherently bound together with uncopyrightable ideas". In other words, the criteria that were chosen to mark the declaratory code of the API as an exception are, on the contrary, quite common for code. Future jurisprudence will have a hard time determining which software is prevailingly functional and which is creative.

Regarding the third factor, copying of code for training purposes is certainly not partial or circumscribed. But US courts already rejected the idea that copying the entirety of a work is incompatible with fair use[17]. Common orientation is that complete copying can be considered fair use when it was "reasonably appropriate to achieve the copier's transformative purpose and was done in such a manner that it did not offer a competing substitute for the original"[18].

About the fourth and final factor, there are grounds to argue that the existence of an AI, capable of auto-compiling and reordering code, does not damage the market for or lessen the value of the original authorial work. After all, this is about code that, even if licensed, was freely accessible by every GitHub user. In *Google Books*, the market/value impact of TDM was justified, concluding that copyright does not include an exclusive right to furnish information about protected works. In the case of open-source code, there are even more reasons to avoid protecting knowledge and insights that can be gained from it. But, even if no loss of value is incurred when training AIs, the same cannot be stated about the machine learning output.

Having weighted all four factors, it feels safe to assume that an US court would consider TDM in the Copilot case fair use. Even so, given that fair use is applied on a case-by-case basis, other instances of TDM might receive a different evaluation.

Aside from fair use, there is another important point that GitHub/Microsoft can underline to justify their copying of the code in the repositories. In fact, according to GitHub's terms of service, hosting content on their platform comports granting a license to them. Following the

---

[15] "So even though Google's use was a commercial endeavor […] that is not dispositive of the first factor, particularly in light of the inherently transformative role that the reimplementation played in the new Android system."

[16] P. N. LEVAL, *Toward a Fair Use Standard*, 103 *Harv. L. Rev* 1105, 1126 (1990). "Copyright seeks to maximize the creation and publication of socially useful material. […] protection is not withheld from authors who lie, cheat, or steal to obtain their information."

[17] P. GOLDSTEIN, *Copyright's Commons*, *29 Colum. J.L. & Arts 1*, 5-6 (2005); *Authors Guild v. Google, Inc.*, No. 13-4829 (2d Cir. 2015).

[18] *Bill Graham Archives v. Dorling Kindersley Ltd.*, 448 F.3d 605, 613 (2d Cir. 2006); *A.V. ex rel. Vanderhye v. iParadigms, LLC*, 562 F.3d 630, 638-640 (4th Cir. 2009).

license's words: "You grant us and our legal successors the right to store, archive, parse, and display Your Content, and make incidental copies, as necessary to provide the Service, including improving the Service over time. This license includes the right to do things like […] parse it into a search index or otherwise analyze it on our servers; share it with other users […]." In this context, "[t]he 'Service' refers to the applications, software, products, and services provided by GitHub, including any Beta Previews"[19]. Given this vague definition of "Service", it can be considered as including the Copilot product.

### 2.2 Solution under European Law

In the European Union, TDM has been the object of a specific provision, with Articles 3 and 4 of Directive 2019/790/EU. Since other countries are making important steps in digital innovation through data analysis, the European Commission recognized the merit of introducing an explicit exception to the right to reproduction. After the Proposal on the 14th of September 2016, the Directive was approved on the 17th of April 2019[20].

Article 2(2) of the Directive defines TDM as "any automated analytical technique aimed at analysing text and data in digital form in order to generate information which includes but is not limited to patterns, trends and correlations". It then proceeds to set out a first exemption in Article 3, but its beneficiaries are only "research organisations and cultural heritage institution".

More interesting for us is Article 4 of the Directive, which again provides an exception for the reproduction and extraction of lawfully accessible works. Everyone can be the beneficiary of this exception, but at the same time rightsholders can expressly reserve the use of their works "in an appropriate manner, such as machine-readable means in the case of content made publicly available online". In other words, rightsholders have the right to opt-out from mining activities. It is also worth noting that the object of this article explicitly encompasses even the works described by Article 4(1)(a) and (b) of Directive 2009/24/EC, meaning "computer programs".

According to the Directive, publishing code on a public GitHub repository, and consequently licensing the software to GitHub, can be also interpreted as allowing mining on the published code. GitHub can argue that, if a programmer does not want to see his work used to train an AI, he should make the repository private or delete it. But the matter is not so simple, since it often happens that programmers add to their repositories even code written by other people, code that they found outside GitHub. Normally, this would not be a problem, since open-source mechanisms allow the free sharing and reuse of code, provided that the correct licenses and notices are included. But Copilot strips code of this essential information, infringing the rights of unknowing third parties.

---

[19] https://docs.github.com/en/site-policy/github-terms/github-terms-of-service.

[20] P. B. HUGENHOLTZ, "The New Copyright Directive: Text and Data Mining (Articles 3 and 4)", *Kluwer Copyright Blog*, July 24 2019, http://copyrightblog.kluweriplaw.com/2019/07/24/the-new-copyright-directive-text-and-data-mining-articles-3-and-4/. C. GEIGER, G. FROSIO, O. BULAYENKO, "Text and Data Mining: Articles 3 and 4 of the Directive 2019/790/EU", *Centre for International Intellectual Property Studies (CEIPI) Research Paper No. 2019-08*, October 17 2019, http://dx.doi.org/10.2139/ssrn.3470653.

Another element of further complication is that the very adequacy of these TDM exceptions is still debated. In fact, it is claimed that the opt-out mechanism will impede the framework development that would permit the EU to catch up with other jurisdictions that adopted much more permissive fair use models towards TDM research. A less aggressive implementation of mining would translate in slower technological progress, that would in turn translate in a loss of market opportunities for European countries. Even limiting these exemptions to works that can be lawfully accessed could mean subordinating mining research to market access. Discrimination of research organisations in reason of their market power could arise, and start-ups would be even more penalized[21].

Additionally, there is no definitive answer on whether AI training is included in the scope of these exceptions. On the topic, in 2021, the European Commission backed a series of surveys and interviews directed at stakeholders and legal experts. A majority of these experts (in this case, most of them were rightsholders or organisations representing rightsholders in the creative sector) stated that the best policy scenario would be to clarify that using data for AI training is not encompassed by the TDM exceptions. Some of the arguments they made are: that AI training functions through exploitation of creative works, partially replacing the actual authors, without giving them proper remuneration and acknowledgment; the possibility that AIs will be trained using European creative works, but would actually benefit companies based outside the EU; the incompatibility with the Berne Convention's three-step test; lack of knowledge on how AI will further develop. Furthermore, advantages were found in the protraction of the status quo, since it will give the chance, to the AI and creative sectors, to keep observing these new conflicts and disturbances that have yet to find a definitive form. Interestingly, 58% of the participants agreed that, during this current status quo, AI companies will assume that AI training is included in the TDM exemptions[22].

This discussion around TDM further extends to the actual scope of protection of the right to reproduction, as harmonized by Article 2 of the Information Society Directive. The more traditional orientation, according to which even ephemeral copies of a purely technical nature are infringing, has been object of critique[23]. This interpretation clashes with how TDM functions, since it is a process that mostly means to extract information from the protected works and (arguably) does not aim to further commercialize them. An antithesis to the technical and literal reading of the right has found its way in the *Pelham* case. In fact, this CJEU decision saw the introduction of a *contextual interpretation* of the right. More in detail, it was ascertained that, even if the ratio for protection of phonographs is the producer's investment (according to recital 10 of the InfoSoc Directive), the determining factor for

[21] GEIGER, FROSIO, BULAYENKO (n 20) 29-37. EUROPEAN COMMISSION, DIRECTORATE-GENERAL FOR COMMUNICATIONS NETWORKS, CONTENT AND TECHNOLOGY, *Study on copyright and new technologies: copyright data management and artificial intelligence*, Publications Office of the European Union, 2022, https://data.europa.eu/doi/10.2759/570559, 216-226.

[22] EUROPEAN COMMISSION (n 21) 204-212.

[23] S. DEPREEUW, *The Variable Scope of the Exclusive Economic Rights in Copyright*, Brussels, Kluwer, 2014, 189 ss ; A. STROWEL, "Reconstructing the reproduction and the communication the public rights: how to align copyright with its fundamentals", *Copyright reconstructed*, 2018, 203.

granting protection is whether the partial reproduction is actually recognisable to the ear[24]. Following an analogous reasoning, in *CV-Online Latvia*, the Court reached the conclusion that it must be concretely ascertained whether an extraction or re-utilisation of contents of a database constitutes an actual obstacle to the database creator's chances of redeeming his past investment[25]. In other words, this emerging contextual approach comports balancing between the interests of authors/producers and those of competitors/users. One of the implications of this approach is that a miner could have grounds to claim that the copies made during the training/input phase do not infringe the right to reproduction if: the protected work is not recognizable when the AI output is considered; and the copy does not impede the author/producer from receiving an appropriate remuneration for the use of the protected work[26]. Then again, the legal problem would become that of ascertaining whether the output coming from the AI constitutes exploitation of the original work.

## 3. Output infringement, to whom belongs the AI output? Can the processing through an AI system clear copyrighted material of intellectual property rights?

### 3.1 Solution under US Law

One of the most relevant claims in the class-action denounces the possible violation of sections 1201–1205 of title 17 U.S.C. as amended by the Digital Millennium Copyright Act. In fact, it can be affirmed that GitHub/Microsoft is distributing a product that circumvents the license system that governs the open-source ecosystem, by stripping code of the Copyright Management Information (CMI, mainly meaning: attribution, copyright notices and license terms).

The official stance of GitHub on this matter is that "[t]he models do not contain a database of code, and they do not 'look up' snippets. Our latest internal research shows that about 1% of the time, a suggestion may contain some code snippets longer than ~150 characters that matches the training set"[27]. Even so, the veracity of these numbers is datable as, in the class-action complaint, were pinpointed multiple instances of verbatim reproduction of code. For example, Copilot can regurgitate fragments of sample code that appears in the online book *Mastering JS*, written by Valeri Karpov; or code similar to the "isPrime" function, that can be

[24] CJEU 29 July 2019, *Pelham GmbH and Others / Ralf Hütter and Florian Schneider-Esleben*, C-476/17, EU:C:2019:624, paragraphs 30, 39.

[25] CJEU 3 June 2021, *CV-Online Latvia / Melons SIA*, C-762/19, EU:C:2021:434, paragraphs 38, 46.

[26] EP (COMMITTEE ON LEGAL AFFAIRS), *Report on intellectual property rights for the development of artificial intelligence technologies* (Oct. 2, 2020), paragraph 10, https://www.europarl.europa.eu/doceo/document/A-9-2020-0176_EN.html; EUROPEAN COMMISSION (n 21) 213-216, "[…] any approach must strike the right balance between the need to protect investments of both resources and effort and the need to incentivise creation and sharing; […] disruptive technologies such as AI offer both small and large companies the opportunity to develop market-leading products […]."

[27] https://github.com/features/copilot.

found in the *Think JavaScript* book by Matthew X. Curinga et al[28]. A possible explanation for this is that programmers, that study on educational books, tend to store their answers on their public repositories, making the original code a recurrent element for Copilot's training. Anyway, what results is that Copilot has been trained and can reproduce (without including the relevant CMI) code belonging to third parties that did not directly publish their work on GitHub. That means that GitHub cannot use its terms of service as a defence.

Section 1201 specifies that nothing in it "[…] shall affect rights, remedies, limitations, or defenses to copyright infringement, including *fair use*, under this title." So, again, this becomes a legal problem of whether the occasional reproduction of licensed content, deprived of its CMI, counts as fair use. The argument would be similar to that of the input issue, but some relevant differences would make it harder to affirm fair use.

In *Authors Guild, Inc. v. HathiTrust*, it was found that the creation of complete digital copies of protected works can be transformative, not when the copies add value or utility, but when they serve "a new and different function from the original work and [are] not a substitute for it"[29]. The verbatim reproduction of copyrighted software as output is not an "extraction of information" like TDM and AI training; it does not offer a different function and it should not be considered a transformative work.

Additionally, the lack of the CMI is not functional for the services that the AI offers. This absence is a choice of the AI's producers, and not an essential part of machine learning. In fact, GitHub could have tried to implement in AI training the necessity of including all relevant information. This shortcoming affects also the fourth factor of the fair use test, since defining what Copilot does as admissible would mean validating all similar tools, capable of bypassing licenses on software, that will multiply exponentially in the next years.

Furthermore, even if bad faith is not a weighty element in fair use assessment, GitHub's narrative should not be ignored. To Copilot's users it is explained that this AI "is a tool, like a compiler or a pen. GitHub does not own the suggestions [it] provides to you. You are responsible for the code you write with [its] help"[30]. In a context where the relationship between AI and copyright still has ambiguities, such clear instructions on output ownership can be reckless, especially when they come from a private company.

Authorship on AI output is a question that has been much discussed in the doctrine, but it still needs a more substantial body of jurisprudence to be further defined. In the worldwide DABUS case, an inventor filed two different patent applications indicating an AI as the inventor. The U.S. Patent and Trademark Office rejected them on the basis of multiple statutory references that described the "inventor" as an "individual". In addition, it affirmed that the act of inventorship requires "conception", of which a machine is incapable[31]. On the other hand, the U.S. Supreme Court defined the author as "the person who translates an idea

[28] M. BUTTERICK, Case 3:22-cv-06823 Document 1, Filed 11/03/22, https://githubcopilotlitigation.com/pdf/06823/1-0-github_complaint.pdf, 18-21.

[29] *Authors Guild, Inc. v. HathiTrust*, 755 F.3d 87, 96 (2d Cir. 2014).

[30] https://github.com/features/copilot.

[31] H. SUN, *Redesigning Copyright Protection in the Era of Artificial Intelligence*, *107 Iowa L. Rev.*, Vol. 107, Issue 3 (March 2022), 1213, 1223-1229.

into a fixed, tangible expression entitled to copyright protection"[32]. Congruently, the U.S. Copyright Office mentioned the creative power of the human mind as indispensable for authorship[33]. In conclusion, it must be concretely evaluated what is the level of human input and supervision on the code that is produced by Copilot.

In an interview, a research intern at Hugging Face evaluated Copilot as inefficient when implementing entire algorithms. The machine needs to be instructed step-by-step and the human programmer still needs to review the code. Copilot performs best when it is asked to complete generic or repetitive code, or to find solutions to small problems that would be normally looked up on search engines or forums[34]. Considering the previous paragraphs of this paper, there are solid grounds to recognize authorship on software that has been adequately vetted by a human. Nevertheless, even if it is true that individual programmers are responsible for the code that they produce with Copilot's help, the machine cannot be fully equated to "a tool like a pen". When it reproduces licensed code without giving its users any kind of warning, the fault can be only of the machine's producers.

Still, in order to find a violation of DMCA §§ 1201–1205, a Court must take in consideration the substance of the reproduction. Even a small amount of copying can be considered outside the scope of fair use when the copied fragment represents the "heart" of the original's authorial expression[35]. On the other hand, the particularities of the US jurisprudence's approach to creative elaboration in coding have already been highlighted. For these reasons, this class-action has the chance of further developing the discussion, through the appraisal of specific software samples. In order to fully determine when code is expressive and when instead it is "inherently bound together with uncopyrightable ideas".

### 3.2 Solution under European Law

The DABUS case was gauged also by the European Patent Office. In this occasion the applications were rejected because it was asserted that the legal framework of the European Patent Convention requires inventors to be natural or legal persons[36]. Even the European stance on authorship is similar to the American one, given that the 2020 EU report on IP and AI describes the principle of originality as "linked to a natural person", and that "the concept of 'intellectual creation' addresses the author's personality" [37].

Consequently, even in the European system programmers must use their own personal capabilities to define the final form of what is outputted by Copilot, supervision and revision. While infringement of the right to reproduction on Copilot's side must be also assessed in concrete.

In the EU, to determine the presence of originality it is "both necessary and sufficient that the subject matter reflects the personality of its author, as an expression of his free and creative

[32] *Cmty. for Creative Non-Violence v. Reid*, 490 U.S. 730, 737 (1989).
[33] U.S. COPYRIGHT OFF., *Compendium of U.S. Copyright Office Practices* §§ 306, 313.2 (3d ed. 2021).
[34] B. DICKSON, "GitHub Copilot is now public — here's what you need to know", *VentureBeat*, June 29 2022, https://venturebeat.com/ai/github-copilot-is-now-public-heres-what-you-need-to-know/.
[35] *Harper & Row, Publishers, Inc. v. Nation Enterprises*, 564–565.
[36] SUN (n 31) 1221-1223.
[37] EP (COMMITTEE ON LEGAL AFFAIRS) (n 26) paragraph 16.

choices"[38]. The Court of Justice also determined that originality might derive from "the choice, sequence and combination" elements that, scrutinized in isolation, are not new or original; as in *Infopaq*, the arrangement of just eleven words was considered enough[39]. Additionally, since "works of applied art" are mentioned in Article 2(1) of the Berne Convention, the practical or utilitarian function of a work should not impede its copyright protection[40]. At the same time, there will be no copyright protection when the expression of a work is dictated only by a technical function since, in such instances, "the different methods of implementing an idea are so limited that the idea and the expression become indissociable"[41]. Following this orientation, a method to demonstrate presence of creative freedom is to prove that "technical considerations, rules or other constraints" were not so prevalent that they left the author no choice on the work's expression[42]. On a slightly different note, in the *Brompton* case, the CJEU concluded that there can be copyright protection, even when the work has been directed by technical considerations, provided that the author was not prevented "from reflecting his personality in that subject matter, as an expression of free and creative choices"[43].

All this considered, there are important elements in the EU Jurisprudence that can bring to the conclusion that even reduced fragments of code can be eligible for copyright protection. Nevertheless, it must be gauged case-by-case how much the programmer added his personal touch to the code arrangement.

An additional element that must be considered is which interpretation of the right to reproduction will be followed in the next years. A contextual reading of it could open new uncertainties. Even so, there are grounds to affirm that reproduction of code without CMI cannot be considered an ephemeral reproduction (even if partial), since it is recognisable and on a systematic scale is capable of damaging the investments and internal rules of the whole open-source community.

## 4. Conclusion. A future all about policy decisions

The GitHub Copilot case and all similar cases that will arise in the next years will be a testing ground for different countries. The US system will have to decide whether an implementation of fair use that strongly favours scientific progress and market development is actually

[38] CJEU 1 december 2011, *Eva-Maria Painer / Standard VerlagsGmbH*, EU:C:2011:798, paragraphs 88, 89 and 94; CJEU 12 september 2019, *Cofemel / G-Star Raw CV*, EU:C:2019:721, paragraph 30.

[39] CJEU 16 July 2009, *Infopaq International A/S/ v. Danske Dagblades Forening*, EU:C:2009:465, paragraph 45. This is also valid for the choice and arrangement of words in a technical document (in the *SAS Institute* case, it was a user manual accompanying software). Creativity is expressed through "choice, sequence and combination" of the elements that compose a work. CJEU 29 November 2011, *SAS Institute Inc. v. World Programming Ltd.*, EU:C:2011:787, paragraphs 66-70, 120.

[40] EUROPEAN COMMISSION (n 21) 153-155.

[41] CJEU 22 December 2010, *Bezpečnostní softwarová asociace Svaz softwarové ochrany / Ministerstvo kultury*, EU:C:2010:816, paragraphs 48-50. This case regarded the graphic user interface (GUI) of computer games.

[42] CJEU 12 September 2019, *Cofemel / G-Star Raw CV*, EU:C:2019:721, paragraph 31.

[43] CJEU 6 February 2020, *Brompton Bicycle Ltd. / Chedech/Get2Get*, EU:C:2020:79, paragraph 26.

justified when it ends up harming private creators. If such a road is followed, it will become even more apparent how easily authorial works with functional elements might end up unprotected.

On the other hand, the European system will have to decide if protecting authorship is more important than catching up with other countries. At the moment, EU Jurisprudence seems on the fence, with emerging orientations, like that of *Pelham*, indicating a desire to make protection of reproduction less rigid.

These questions are particularly pressing when it comes to new technologies. In the 2019 *Dreamwriter* case, a Chinese Court recognized as copyrightable a financial reporting article that was written by an AI. It was considered that the formal requirements of a literary work were met because of the data arrangement and selection undertaken by the Tencent human team that worked with the machine[44]. So, even if autonomous AI authorship will probably never be recognized, there are many jurisdictions that are taking bolder steps on this matter. Will the US and EU focus on protecting authorial rights or will they be more preoccupied with not being left behind? Both answers have weighty implications.

Going back to the Copilot case, it has the potential of becoming one of the future hubs where these policy decisions will be made. With the warning that, in this instance, the open-source ecosystem could be severely damaged. As a closing note, it has to be reported that GitHub recently claimed its intention of introducing in 2023 a function to identify the Copilot's output strings matching public code, providing a reference to those repositories[45].

[44] SUN (n 31) 1218-1219.

[45] R. J. SALVA, "Preview: referencing public code in GitHub Copilot", *GitHub Blog*, November 1 2022, https://github.blog/2022-11-01-preview-referencing-public-code-in-github-copilot/.