COPYRIGHT CONUNDRUMS IN GENERATIVE AI: GITHUB COPILOT'S NOT-SO-FAIR USE OF OPEN-SOURCE LICENSED CODE

VIVEK BASANAGOUDAR* & ABHIJAY SREKANTH**

ABSTRACT

Generative AI has made ground-breaking developments in multiple industries worldwide. One of its use-cases in the software industry is that of GitHub Copilot. Trained on billions of lines of publicly available code, GitHub Copilot assists in software development but has been subject to claims of copyright infringement. By reproducing original code in snippets as output, the tool has been accused of violating terms of several open-source licenses by failing to provide any form of attribution. OpenAI, GitHub and Microsoft, in their defence, have relied on the fair use doctrine and its applicability in Google v. Oracle without paying heed to the stark difference between declaring code and implementing code in open-source licenses. Using the GitHub Copilot dispute as a contextual backdrop, this article seeks to examine copyright infringement in open-source licensed code and thereafter determine whether 'training ML systems on public data' and subsequent reproduction of code by generative AI is fair use. To substantiate, it shall also briefly discuss the significance of striking a balance between economic harm and moral rights, the difference between declaring code and implementing code and its relevance to the fair use doctrine, and the proposed copyright rules in the EU AI Act.

I. INTRODUCTION

Artificial Intelligence ["**AI**"] has emerged as a fine competitor to its human counterpart by progressively achieving feats ranging from defeating experts at Chess and Go[1] to scoring in the 90th percentile of the bar examination.[2] Although neural networks of AI have not yet reached the functionality of a human brain in terms of awareness, imagination, inventiveness and creativity,[3] a determined pioneer is certainly paving the way. OpenAI, a combination of a non-profit AI research

* Vivek Basanagoudar is a final-year law student at O.P. Jindal Global University.

** Abhijay Srekanth is a master's student at Queen Mary University London specialising in Intellectual Property law.

[1] Peter Van Der Made, *The Future of Artificial Intelligence*, FORBES (Apr. 10, 2023), https://www.forbes.com/sites/forbestechcouncil/2023/04/10/the-future-of-artificial-intelligence/?sh=6750821a4ac4.

[2] Karen Sloan, *Bar exam score shows AI can keep up with human lawyers, researchers say*, REUTERS (Mar. 15, 2023), https://www.reuters.com/technology/bar-exam-score-shows-ai-can-keep-up-with-human-lawyers-researchers-say-2023-03-15/#:~:text=GPT%2D4%2C%20the%20upgraded%20AI,of%20legal%20technology%20company%20Casetext.

[3] J. Doe 1 & J. Doe 2 v. GitHub, Inc. & Ors., No. 21-cv-06823, complaint at 26 (N.D. Cal. filed Nov. 03, 2022), https://githubcopilotlitigation.com/pdf/06823/1-0-github_complaint.pdf.

laboratory and a for-profit subsidiary, stands at the forefront of AI innovation owing to its diverse understanding of machine learning, and subsequent application to speech recognition, art creation, gaming and math, among many other fields.[4]

Over 300 million jobs and 18% of work globally are said to be exposed to automation in the near future.[5] Legal, Office and Administrative Support, and Architecture are the first of several U.S. employment industries to bear such a risk.[6] With automation disrupting multiple commercial sectors, developing a system to automate code development was extremely tempting. Taking steps in that direction, GitHub (an online cloud-based code-hosting platform) and OpenAI collaborated to launch the '*world's first at-scale generative AI development tool*'[7] called GitHub Copilot ["**Copilot**"]. GitHub cop-pilot leverages AI developed by GitHub and OpenAI to automatically generate code in a manner similar to predictive text functions on Microsoft word, email software systems, etc. GitHub Copilot claimed its software could increase the pace of coding by 55%.[8]

Having been trained on '*billions of lines of public code*',[9] Copilot has faced criticism for copyright infringement[10] and is currently being sued. Two anonymous coders ["**plaintiffs**"] brought multiple claims of copyright infringement, breach of contract, privacy and competition law concerns against GitHub, OpenAI and Microsoft (an investor in both corporations) ["**defendants**"]. Focusing on copyright infringement, the plaintiffs have contended an infraction of multiple open-source licenses through Copilot's training on, and consequent reproduction of, their code. The commercialisation of code authored by the plaintiffs and coders worldwide has resulted in a class-action lawsuit with a focus on the lack of attribution.[11] In essence, the plaintiffs claim that defendants commercially worked the plaintiffs open-source code made available without adhering to the terms of the license, thereby infringing their copyright in the code they created. In a very discreet fashion, the defendants suggested that the fair use doctrine would negate copyright

[4] *Research Index*, OPENAI (last visited May 03, 2023), https://openai.com/research.

[5] JOSEPH BRIGGS & DEVESH KODNANI, GLOBAL ECONOMICS ANALYST: THE POTENTIALLY LARGE EFFECTS OF ARTIFICIAL INTELLIGENCE ON ECONOMIC GROWTH (2023).

[6] *Supra* note 5.

[7] Thomas Dohmke, *GitHub Copilot X: The AI-powered developer experience*, THE GITHUB BLOG (Mar. 22, 2023), https://github.blog/2023-03-22-github-copilot-x-the-ai-powered-developer-experience/.

[8] Eirini Kalliamvakou, *Quantifying GitHub Copilot's impact on developer productivity and happiness*, THE GITHUB BLOG (Sep. 7, 2022), https://github.blog/2022-09-07-research-quantifying-github-copilots-impact-on-developer-productivity-and-happiness/.

[9] Albert Ziegler*, GitHub Copilot Research Recitation*, THE GITHUB BLOG (Jun. 30, 2021), https://github.blog/2021-06-30-github-copilot-research-recitation/.

[10] *Analyzing the Legal Implications of GitHub Copilot*, FOSSA (Jul. 14, 2021), https://fossa.com/blog/analyzing-legal-implications-github-copilot/.

[11] Matthew Butterick, *We've filed a lawsuit challenging GitHub Copilot, an AI product that relies on unprecedented open-source software piracy*, GITHUB COPILOT LITIGATION (Nov. 3, 2022), https://githubcopilotlitigation.com/.

infringement claims. As AI based systems are significantly dependant on the training data used[12], legal jurisprudence on the manner in which data use is copyright compliant would have massive implications on future development practices.

II. Copyright Infringement in Open-Source Licenses: Balancing Economic Harm and Moral Rights

Briefly put, open-source licenses have been integral to the development of Free and Open-Source Software (FOSS). Open-source licenses can be categorised into copyleft or permissive licenses depending on whether modified works are offered along the same terms as the initial license. The GNU General Public Licenses [“**GNU GPL**”] Family (GPL 1.0, 2.0 and 3.0) are prominent copyleft strategies devised by Richard Stallman which ensure that code is made available to study, modify and share easily. When a coder lifts code governed by a GNU GPL license, then upon distribution, it must be done in a manner identical to the original license without additional restrictions while also outlining the original source code, thereby ensuring that such code would perpetually be available and not barred in a proprietary manner.[13] This may include, depending on the nature of the GNU GLP license, a requirement to make the independent work developed by the licensee available on the same GPL GNU license terms when changes are made to the code licensed on a GPL GNU public license. This particular license will be discussed further later on in this section. The MIT License is a modern permissive license which mandates only sharing of a copyright notice and the license itself – *‘The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.’*[14] The stark difference between the two lies in the ability to make modified works proprietary, but the similarities of open-source licenses – which require licensees to reproduce a copy of the license, source code, copyright notice and author details[15] – are more relevant to the Copilot copyright infringement claims.

Before delving into the copyright infringement claims, it is imperative to determine if violation of open-source licenses fall within the contractual or copyright domain. The plaintiffs in the present case have contended both breach of contract and copyright infringement under the Digital Millennium Copyright Act [“**DMCA**”].[16] The most-cited decision on the subject is that of the United States Court of Appeals for the Federal Circuit in *Jacobsen v. Katzer*.[17] The dispute centred

[12] *Id.*

[13] Catharina Maracke, *Free and Open Source Software and FRAND-based patent licenses, How to mediate between Standard Essential Patent and Free and Open Source Software*, 22 J. WORLD INTELLECT. PROP. 78-102 (2019).

[14] *The MIT License*, OPEN SOURCE INITIATIVE (2018), https://opensource.org/licenses/MIT/.

[15] *Supra* note 12 at 5.

[16] Digital Millennium Copyright Act, 17 U.S.C. §§ 512 (1998).

[17] Jacobsen v. Katzer, 535 F.3d 1373 (Fed. Cir. 2008).

around the use of software that Professor Jacobsen had made available to the public. The downloadable files he made available contained a copyright notice which referred users to a 'copying file' which set out the terms of the artistic license. The terms of the license allowed any user the right to copy, modify, and distribute the software, provided that the user prominently provided a notice where and when they changed one or more files in the software provided by Professor Jacobsen, and also make the modification freely available.

In *Jacobsen*, Mathew Kazter, and his company Kamind Associates used the bundle of software that Professor Jacobsen had made available to distribute their own software. However, they failed to abide by the terms of the license, i.e., make parts modified publicly and freely available. Professor Jacobsen therefore applied for a preliminary injunction against Katzer before the District Court. The District Court noted that the copyright had been licensed to Katzer under a non-exclusive open-source license. It relied on *Effects Associates Inx v. Cohen*[18] which held that a copyright license waived any claims of suing for copyright infringement, but not breach of contract. It therefore held that Katzer's use was a breach of contract, not breach of copyright.

Professor Jacobsen raised an appeal before the United States Court of Appeals for the Federal Circuit. The court on examining the text of the license noted that the district Court had erred in its approach and reversed the lower court's decision. It noted that the license expressly governed the manner in which the copyrighted code was to be downloaded, modified, distributed and in any other way commercially exploited. It expressly noted that "*copyright holders who engage in open-source licensing have the right to control the modification and distribution of copyrighted material.*"[19] It placed reliance on the Second Circuit Court's decision in *Gilliam v. ABC*[20] which held that unauthorised editing of work, as was the case with Katzer's code, would amount to copyright infringement when such editing takes place outside the scope of the license.

Interestingly, and especially important for the purposes of this article, the court went on to note how there was in fact a violation of Katzer's economic rights through a violation of the open-source license. The artistic license governed the manner in which creators were to be attributed, increase the rate at which innovation in respect of a project was made publicly available and increase awareness of said project by driving traffic towards the open-source incubation page. The

[18] Effects Assocs., Inc. v. Cohen, 908 F.2d 555 (9th Cir. 1990).
[19] *Supra* note 17.
[20] Gilliam v. ABC, 538 F.2d 14 (2d Cir. 1976).

harm to potential technical advances as a result of failure to disclose was sufficient to amount to a form of economic harm.

Taking note of the precedent set for economic harm in copyright infringement of open-source licenses, the defendants' response asserts that the plaintiff's complaint *'fails on two intrinsic defects: lack of injury and lack of an otherwise viable claim'*.[21] They contend that the plaintiffs do not *'advance a copyright infringement claim'*[22] and that no harm has been caused by Copilot. However, based on the ruling, violation of open-source licenses certainly falls within the domain of copyright infringement and more importantly, harm to potential technical advances by failure to disclose would constitute economic harm. Copilot, by reproducing blocks of open-source licensed code, fails to disclose the author details, copyright notices and the license itself, and has the potential to harm technical advances if such works are not attributed in the described manner. The lack of attribution is exactly what constitutes violation of terms of the relevant open-source licenses and has been contended by the plaintiffs to cause economic harm, regardless of quantifiability. The plaintiffs also managed to account for quantifiability in terms of infringement by using GitHub's own data which stated that about 1% of the times, a snippet is matched. Based on its user base, the plaintiffs estimated over 36,000 DMCA violations which could be considered as substantial economic harm to coders, considering all reproduced open-source licensed code was commercialized when Copilot offered a paid subscription model.[23]

Even if one were to consider economic harm to be determined individually based on each license, a common issue arising for all would be the lack of attribution as a violation of moral rights. The defendants repeatedly negate copyright infringement claims on the grounds of lack of injury, which may hold good when considering commercial value, but are not a necessity for moral rights. Article 6bis of the Berne Convention for the Protection of Literary and Artistic Works ["**Berne Convention**"] mandates minimum standards for moral rights in member states. It accounts for economic rights and their transferability but still guarantees the right to attribution/paternity right to an author by stipulating the following – '*independently of the author's economic rights, and even after the transfer of the said rights, the author shall have the right to claim authorship of the work*'.[24] Moral rights were

[21] GitHub & Microsoft Corporation's Memorandum in Support of Its Motion to Dismiss, J. Doe 1 & J. Doe 2 v. GitHub, Inc. & Ors., No. 21-cv-06823, at 10 (N.D. Cal. Jan. 26, 2023), https://fingfx.thomsonreuters.com/gfx/legaldocs/byprlrzkrpe/MICROSOFT%20OPENAI%20LAWSUIT%20microsoftmtd.pdf.

[22] *Id.*

[23] *Supra* note 12 at 27.

[24] Berne Convention for the Protection of Literary and Artistic Works, art. 6bis, Sep. 9, 1886, as revised at Paris on July 24, 1971, 1161 U.N.T.S. 3.

intended to be conferred on all works, but the U.S. has received scrutiny for only partially complying and awarding them to 'visual arts' under the Visual Artists Rights Act.[25] Code does not fall within the purview of this Act and hence, has resulted in moral rights not being discussed by either the plaintiffs or defendants in this dispute. However, if a similar suit is brought in a different jurisdiction,[26] it will inevitably serve as the most basic contention (particularly in light of the proposed copyright rules of the European Union ["**EU**"] AI Act, which shall be discussed later).

For an example of how GitHub Copilot reproduced open-source code without proper attribution, examine the lines of code below-

```
'function isEven(n) {
if (n == 0)
return true;
else if (n == 1)
return false
else if (n < 0)
return isEven(-n);
else
return isEven(n - 2);
}
console.log(isEven(50));
// → true
console.log(isEven(75));
// → false
console.log(isEven(-1));
// → ??'[27]
```

The abovementioned code is a response secured from OpenAI's Codex on which Copilot functions. The prompt provided was *'function isEven(n)'*[28] based on which Codex assumed that the code was trying to test if a number was even. The six additional lines after the function, starting with *'console.log(isEven(50))'*, are tests to check if the code provides correct answers. All of this code

[25] Visual Artists Rights Act, 17 U.S.C. §§ 106A-106C (1990); *Moral Rights in U.S. Copyright Law*, COPYRIGHT LAWS (Jan. 08, 2023), https://www.copyrightlaws.com/moral-rights-in-u-s-copyright-law/.

[26] MIRA T. SUNDARA RAJAN, *The Programmer as Author: Moral Rights in Information Technology in Moral Rights: Principles, Practice and New Technology,* OXFORD UNIVERSITY PRESS, NEW YORK, 283, 320 (2011).

[27] MARIJN HAVERBEKE, ELOQUENT JAVASCRIPT: A MODERN INTRODUCTION TO PROGRAMMING (No Starch Press, 3rd ed. 2018).

[28] *Id.*

has evidently been picked up from *'Eloquent Javascript'* by Marijn Haverbeke because the last two lines of code are actually placeholder values that include the *'??'* error. Instead of fixing the issue, Codex has verbatim reproduced all ten lines of the function as well as six lines of tests. If a coder were to receive this output, no copyright notice, no license and no attribution would be provided to Marjin Haverbeke, constituting a violation of the terms of open-source licenses and in other countries, moral rights.

Economic harm and moral rights are to be balanced equally, which does not seem to be the case in the U.S. currently. Such a grave lacuna enables parties like the defendants to merely claim lack of injury as a counterargument to copyright related matters. If the right to attribution/paternity right was conferred on all works, as is the case in several other countries,[29] coders would not be burdened with proving quantifiable violation of their economic rights or having to argue the lack of attribution solely based on it being documented in an open-source license.

III. LACK OF TRANSFORMATION VIS-À-VIS THE FAIR USE DOCTRINE: DECLARING CODE V. IMPLEMENTING CODE

For the sake of an arguendo, the defendants presumed that Copilot's training data was based on copyrighted material and reproduced the same as output.[30] They contended that if a copyright infringement suit did persist, it would be subject to *scenes à faire*, the merger doctrine and the fair use doctrine ["**fair use**"].[31] However, placing reliance on *Google v. Oracle*[32] (as was done by the defendants) may not fulfil the requirements of fair use as the open-source licensed code in question is of an entirely different kind.

Fair use has been defined as *'any copying of copyrighted material done for a limited and "transformative" purpose'*[33] and does not warrant the need for permission from the copyright owner. Over time, the U.S. has recognised several types of fair use including but not limited to comments, criticism and parodies of works. *Campbell v. Acuff-Rose Music, Inc.*[34] saw the birth of transformative use which considered whether the new work was of an entirely different purpose from the original. The landmark judgement established the route ahead as its analysis of transformative use is heavily

[29] *Supra* note 25.

[30] *Supra* note 21 at 20.

[31] *Id.*

[32] Google LLC v. Oracle America, Inc., 141 S. Ct. 1183 (2021).

[33] Richard Stim, *What is Fair Use?*, STANFORD UNIVERSITY (May 04, 2022), https://fairuse.stanford.edu/overview/fair-use/what-is-fair-use/.

[34] Campbell v. Acuff-Rose Music, Inc., 510 U.S. 569 (1994).

relied on when reading Section 107 of the Copyright Act, 1976 which laid down four factors to determine fair use[35] –

> *(1) the purpose and character of the use, including whether such use is of a commercial nature or is for non-profit educational purposes;*
> *(2) the nature of the copyrighted work;*
> *(3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and*
> *(4) the effect of the use upon the potential market for or value of the copyrighted work.*

Authors Guild, Inc. v. Google, Inc.[36] relied on this precedent and ruled in favour of Google for digitising a few pages of copyrighted books since *'words in books are being used in a way they have not been used before'*. It was transformative and served a useful purpose without causing any harm to the original works. Both the cases ruled that the aforementioned factors were to be weighed equally and were not necessarily exhaustive in nature.[37]

In the realm of open-source licensed code, a monumental decision was pronounced two years ago in *Google v. Oracle*. Deliberating on Application Programming Interface code, the U.S. Supreme Court unanimously held that Google's usage of Oracle's code was fair use. However, this decision was marked with a contingency of splitting the API code into two types – declaring code and implementing code. Declaring code was understood by the Court to be code that declares the name of the method along with its inputs and outputs.[38] Implementing code was understood as instructions that carry out the function of the particular method. In determining Google's fair use, the Supreme Court came to the very logical conclusion that declaring code could be used by coders as long as the purpose was transformative, and the transformation of code was conducted by reimplementation (writing different implementing codes). Some scholars have also argued along the same lines by asserting that coders writing their own implementation codes after borrowing declaring codes would be considered transformative use and thereby, would be permitted.[39]

```
'function isEven(n) {
if (n == 0)
```

[35] Copyright Act, 17 U.S.C. § 107 (1976).

[36] Authors Guild, Inc. v. Google, Inc., 804 F.3d 202 (2d Cir. 2015).

[37] Michelle L. Childs, *Defending the Digital Millennium Copyright Act: The Role of the Anti-Circumvention Provisions in the Copyright Law*, 5 J. INTELL. PROP. 142 (2005).

[38] Jeffrey Robert Kaufman, *What Google v. Oracle means for open source*, OPEN SOURCE (May 05, 2021), https://opensource.com/article/21/5/google-v-oracle.

[39] *Id.*

```
return true;
else if (n == 1)
return false
else if (n < 0)
return isEven(-n);
else
return isEven(n - 2);
}
console.log(isEven(50));
// → true
console.log(isEven(75));
// → false
console.log(isEven(-1));
// → ??’[40]
```

Relying on the same example, *‘function isEven(n)’* serves as the declaring code as it declares the method which is to find out if a number is even and it also declares the integer *‘n’* as the input. All the following code from *if (n == 0)* to *return isEven(n - 2);}* serves as the implementing code. The *‘console.log(isEven(50));’* code are tests and are considered programmer inputs and method calls which fall outside the category of declaring code and implementing code.

The defendants have claimed defences of merger doctrine and fair use in a blanket-manner for usage of all open-source licensed code by relying on *Google v. Oracle* and other judgements. The bifurcation provided by the Supreme Court has been completely ignored which could set a dangerous precedent by failing to protect implementing code, if accepted. Declaring code and the method call tests are logically to be affected by the merger doctrine since they can only be expressed in one form alone. No matter the modifications made, if one integer such as *‘n’* or a specific number such as *‘50’* is to be used, the expression will largely remain the same. It is interesting to note that the Supreme Court answered the question of fair use by Google after assuming that declaring code could be copyrighted. Such an assumption was made to avoid delving into the merger doctrine (which would answer whether APIs were copyrightable).[41]

Fundamentally, copyright exists to incentivise creativity and protect authors rights over their works.[42] Coders exhibit the required creativity in generating a declaration that signals an operation,

[40] *Supra* note 26.

[41] *Supra* note 37.

[42] Schroff S, *‘The Purpose of Copyright—Moving beyond the Theory’*, 16 J. INTELL. PROP. L. 1262 (2021).

as well as in designing implementing code that carries out such an operation. Implementing code seems to be a unique expression of logic created by a coder and would certainly be protected by copyright. Several functions could possess different implementing codes, thereby permitting fair use of the declaring code since there is a transformative purpose every time the implementing code is changed. Coders may also arrive at the same solution through a completely different implementing code which makes the expression unique. While transformative use, as per *Google v. Oracle*, would allow for declaring code to be copied, directly lifting implementing code and reproducing it would not be permissible due to a lack of transformation and the second factor of '*the nature of the copyrighted work'* not being satisfied. Hence, reproducing Marjin Haverbecke's code from *'Eloquent Javascript'* would only be permissible for the declaring code, not the implementing code and would still constitute copyright infringement of open-source licensed code.

IV. Concluding Remarks: A Glimmer of Hope in the EU AI Act

U.S. copyright legislation awaits a much-needed reform in its system of granting moral rights. In the absence of moral rights protection, authors of literary works like code who try ensure attribution through the terms in their open-source licenses have no real recourse in the US to demand attribution outside of these licenses. The credit-problem may perhaps only be worsened by increase in use of generative AI. This is in large part due to the immense sets of training data that are inputted, with arguably lower attention to the terms of use regarding such datasets.[43] . This is largely based on two factors- i) attribution is essential for small developers and artists to develop a larger following; and ii). identifying economic harm where the role of a piece of code is minor in an AI-generated work may be tenuous. Further, if fair use is granted to all forms of code, then infringement claims would be redundant in courts and even the current remedy for attribution would fail.

The defendants may have some remedies considering that the Copilot case was filed in the U.S. However, as previously mentioned, if filed in other jurisdictions, the lack of attribution would not be taken as lightly and would bear greater implications. Moral rights were envisioned to provide the bare minimum recognition to authors for their contribution and could be claimed

[43] Several lawsuits against AI developers have recently been filed, claiming breach of terms of use. See Mia Sato, *Drake's AI Clone Is Here — and Drake Might Not Be Able to Stop Him*, THE VERGE (2023), https://www.theverge.com/2023/5/1/23703087/ai-drake-the-weeknd-music-copyright-legal-battle-right-of-publicity (last visited Sep 13, 2023); Dan Milmo & Dan Milmo Global technology editor, *Sarah Silverman Sues OpenAI and Meta Claiming AI Training Infringed Copyright*, THE GUARDIAN, Jul. 10, 2023, https://www.theguardian.com/technology/2023/jul/10/sarah-silverman-sues-openai-meta-copyright-infringement (last visited Sep 13, 2023).

independently of economic rights and their transfer. Evident from the proposed copyright rules for generative AI in the EU AI Act,[44] disclosure and eventually, attribution will be on the rise. The proposed copyright rules require companies deploying generative AI to disclose any copyrighted material used in their training and development.[45] While this may not directly mandate reproduced code to provide attribution, it is a step in the right direction and coupled with enforceability of open-source licenses and moral rights, is likely to create more transparency in the functioning of generative AI.

[44] Supantha Mukherjee et al., *EU proposes new copyright rules for generative AI*, REUTERS (Apr. 28, 2023), https://www.reuters.com/technology/eu-lawmakers-committee-reaches-deal-artificial-intelligence-act-2023-04-27/.
[45] *Id.*