A Critical Review of Open Source Software Development: Freedom or Benefit Libertarian View Versus Corporate View

Rizwan Ahmad, *University of Sydney Business School*

This is a critical literature review of the open-source software landscape and what purpose it aims to serve by accelerating a debate in the field of IS about what open source means and how it has transformed the social and economic landscape of the software industry. As the general population has developed an increased dependence on information technology, the necessity for quality software offerings has also increased. This article attempts to present, interpret, and compare the two contrasting theories of the open-source landscape—the libertarian view versus the corporate view. Subsequently, it aims to delve into the battle of these two conflicting theories and what their interpretations present from the perspective of socially embedded lens—freedom or benefit (profit). This article will examine the differing opinions and dichotomy of the scholars from multiple angles and if the juxtaposition of these two theories creates the holistic view of the landscape.

Software development had humble originations as a means to provide collaboration between scientists and engineers, but as information technology has established a pervasive presence throughout the world, the development of software has emerged as a major industry. The software serves the purpose of providing the instructions for the computer and subsequent programs to work correctly and many corporations, such as Microsoft and Apple, have established a firm grasp over the software market. As a result, the emergence of open-source software has presented a challenge to the traditional offerings by providing free alternatives. Open-source software developments are basically Internet-based communities that voluntarily collaborate in developing software on the Internet and such Internet communities have become an important cultural and economic phenomenon. As the corporate environments, in the form of closed source, limit the way in which technologies can empower the oppressed and marginalized, the open-source actors have much more freedom in proactively modifying, shaping, and reshaping such technologies, both in terms of its use and design. Open source gained momentum in 1998 under the moniker "free software" and can be defined as "a method and philosophy for software licensing and distribution designed to encourage use and improvement of software written by volunteers by ensuring that anyone can copy the source code and modify it freely" [1]. However, Bollinger[2] argues that an open source is a free software and "gives users the right to run, copy, distribute, study, change, and improve it as they see fit, without having to ask permission from or make fiscal payments to any external group or person" (pp. 2). Hence, based on the critical theory of technology or/and philosophy of technology as best represented by Andrew Feenberg that cogently deals with intertwined nature of the technical and social,[3] open-source model cannot be separated from "social conscious" and may be able to bring a positive change in the field of software development.

This has resulted in debates over the open-source software that has been going on for long and scholars have pondered over the validity and quality of open-source software and what motivates the coders to engage in such a practice. Before we delve into a dichotomy of what open software is, we have to get into etymology of the term "open source" and what it means to different scholars. Next, we will discuss and assess what motivates coders to contribute to open-source software development and if it has anything to

1520-9202 © 2021 IEEE
Digital Object Identifier 10.1109/MITP.2020.3014450
Date of current version 29 January 2021.

do with their personal values. Finally, we will discuss how open-source software development model acts as a "societal good" denoting "freedom," and a "business model" denoting "benefit" and whether these two terminologies are "mutually exclusive" based on the perspectives of "libertarians" and "corporate" to be discussed further in the article.

METHODOLOGY

The critical literature review was considered an appropriate methodology to discuss and assess what open-source software means, how it has changed the social, cultural, and economic landscape and how it pertains to "societal good" or/ and a "business model." The first step was to select appropriate publications on open-source software development. Based on the approach of Webster and Watson,[4] we chose the search through leading journals or/and books/book chapters within the field of open source software development. The books or book chapters were also taken into consideration to maintain the versatility of articles chosen. The articles chosen included not just literature, but also case studies and novel empirical findings on open-source software development to explore the research about how the open-source community has evolved over time. For this purpose, only academic sources were considered for literature review and hence industry sources were discarded. The following digital libraries were used to access articles on open source software development.

(i) Wiley InterScience (www.interscience.wiley. com)
(ii) The ACM Digital Library (www.portal.acm.org)
(iii) Google Scholar (www.scholar.google.com)

Initially, 35 articles using the keyword "open source software" from each of the above digital libraries were randomly selected for a total of 105 articles. These articles were chosen from a specified time period between 1996 and 2017. The time period between 1996 and 2017 was considered significant because the hacker culture of open-source software came into existence around 1996, and since the review needed to emphasize on how open-source community has constantly evolved as a result of societal "pull" and "push," it was decided to review literature from the period 1996 until 2017 to assess how the open-source community has evolved and ingrained itself in social, cultural, and economic landscape in the last 21 years. Thereafter, we manually refined the articles on the basis of the "keywords" or/and "abstracts" that coincided with the origin or/and the meaning of open-source software and coincided or related to terminologies, such as freedom, motivations, idealism, business model, reputations, recognition, societal good, corporate view, socialist view, openness, social gifts, and altruism in relation to open-source software development. Finally, using the abovementioned criterion, we selected a total of 55 articles on open-source software development out of which 20 articles were from books or book chapters and the rest were journal articles that included four discussion papers on open-source software research as well. Articles that were not about open-source software development or/and did not meet the abovementioned criterion were discarded.

ANALYSIS

The analysis of the review was conducted on the basis of the following ways.

(i) Statements about open-source software development model and its origin.
(ii) Statements about open-source software development model in relation to societal good, altruism, gifting, idealism, socialism, collaboration, etc.
(iii) Statements about open-source software development model in relation to recognition, reputation, business model, monetary gain, innovation, etc.
(iv) Personal values and motivations involved in contributing to open-source software development model.
(v) Statements about open-source software development model in relation to its socialist or libertarian views and corporate or capitalist views.

In the section libertarian view versus corporate view, I have tried to use Benkler[5] as a reference point because Benkler seems to be a strong proponent of open-source software model for both "societal good" and an efficient "economic or business model."

LITERATURE REVIEW

Origin of Open-Source Software

In 1998, Eric Raymond along with other high profile programmers started Open Source Initiative. Raymond perceived Stallman as a regulatory idealist, who preferred politics to efficiency and technical excellence. A number of programmers perceived Raymond's initiative as efficient and friendly for free software business. Hence, the term "open source" was chosen to circumvent Stallman's political concerns and rhetoric of "free".[6] It must be noted that Stallman

focused on decommodification of the market in relation to software development because it enhanced individuals' ability to use and modify, whereas Raymond focused on organization of production and not on decommodification.[6]

Finally, the origin of the word came into being in 2001 when Eric Raymond[7] published his work titled "The cathedral and the Bazaar." The use of these metaphors in the nomenclature associated with the distribution of software allows one to develop a comprehension of the premises presented while models provide representations that may be partial or incomplete. However, as the concepts identified through the use of metaphors prove to be productive, they then become models. Two primary metaphors applied in this area consist of cathedrals and bazaars to serve as a method of identifying proprietary software and open-source software. Here, "cathedral" represents "centralized planning" and hence points toward a "proprietary" software and indicates a single point focus, where one entity prepares a completed package; whereas "bazaar" represents a new "liberal utopia" with various competing ideas and agenda amalgamating at a much faster speed and hence points to an "open source software" as there are many separate individuals who collaborate freely to produce an item.[7]

SOCIETAL GOOD

The free software movement was founded on the idea that writing the code is an act of free speech. This free speech may be referred to as a dimension of hacker culture and classified as a wider social justice movement that expands beyond hacker culture to disparate social groups worldwide. Hence free or open software movement emanates from a technological trend that criticizes and challenges copyright laws that has gone through numerous variations in relation to the nature of freedom surrounding software development and artistic expressions.[8] Hence, free or open software offers information justice, which is a just and fair rearrangement of social goods that originates from information technology.[9] However, Schmidt and Schnitzer (2003), as cited in the article by Arslan[10] argues that free or open-source software provides freedom to coders, who have the right to use, change, expand, restructure, and reorganize the original source code or modified software. Additionally, "freedom promotes wide utilization and continuous development of innovations"[11] pp. 1371). Hence, it signifies four types of freedom as given below that open-source coders enjoy[10] pp. 158, 57:

(i) The freedom to run the code or program for any reason without restrictions.
(ii) The freedom to investigate how the code or program works and the need to change it, provided the source code is easily accessible.
(iii) The freedom to reallocate copies in order to help your neighbors.
(iv) The freedom to better the codes and release such improvements to public to benefit community as a whole provided the source code is easily accessible.

As the use of free or open-source software by one person does not affect its use by another person and its codes are readily available to public or society, it satisfies the criterion of being a societal or public good but open source is not a traditional societal good. It is rather dynamic, market segmented, complex and is driven by specific consumers' needs and demands. Such needs and demands decide the fate of open-source software and make the product finished that goes through a number of motivating incentives.[12] Hence, it is the opportunity cost of not developing the open-source software (the cost of proprietary software) that motivates individuals/coders to develop it.[12] Agrawal[12] further argues that free riding is not considered an issue with open-source software as it enhances social welfare. He further posits, "In the traditional economy, free riding is something policymakers want to minimize. In this unusual case, free riding may actually improve the efficiency of allocating the public good (by preventing waste) if programmers are sufficiently motivated to provide the good privately in the first place" (pp. 59). Additionally, open-source coders make the internal design of the technology and its beta versions readily available to public so that they can contribute to its development. Hence, open-source software development can be explained as a type of knowledge production and material practice.[13] But free or open software is also a process of technological development based on social collective as envisaged by Kelty[14] based on the notion of recursive public: "a public that is constituted by a shared concern for maintaining the means of association through which they come together as a public" or society (pp. 28).

Hence, we should look for ethics or generalized values in the hacker community of open source movement that promote information sharing as a positive and powerful societal good and hence hackers' ethical duty is to write open source codes and they should always made themselves readily accessible to computing and information resources.[15] However, Moody[16]

provides a more general overview of hackers' ethics in free or open-source movement by using terminologies such as community, openness, sharing, creation, freedom, joy, cooperation, and beauty. Himanen[17] echoes with the same sentiment and recognizes seven values—openness, caring, freedom, social worth, passion, activity, and creativity, and three ethics—work, network, money, and argues that a "new spirit" is slowly unfolding from the hacker community to society at large in the open-source movement. Here, we see that Raymond's definition of hackers' ethics is narrower than Moody's and Himanen's definition of hackers' ethics.

Additionally, "Alumni effect" arguably lowers the cost of collaboration while working in open source landscape as opposed to working in proprietary landscape.[18] This explains why software developers, engineers, and programmers reignited the collaborative measures from the early days of computer technology to present open-source software as a method to combat the increasing costs associated with the total cost of computing, as proprietary software became more prevalent. Consequently, open-source software is defined as a type of programming codes available to public, in this case software coder(s) irrespective of their geography or demography, who can further create or modify codes freely as per their respective requirements, thereby giving the name "open source."[19] However, open source is based on nonproprietary model and on shared contributions for a common project without having any right for anybody to be excluded from part of the project or the whole of it.[5] But, open-source software does not mean "free" in terms of money or in terms of making the payment but it means "freedom" or "free speech."[20] This explains why open-source software is argued to be a "social" and "economic" phenomenon and comprises of software developers, who collaborate to develop or exchange software codes without expecting any monetary gain, which they can, both individually or/an collectively, freely builds on each other's or modify and distribute freely for use or for further modification.[21] Hence open source is perceived as a collaborative community, which does not exclude anybody from enhancing, modifying, and remodifying the codes, nor does it allow anybody to exert proprietary claims to any part of the codes he/she/they have developed.[22] Hence, we can see that scholars have used the term "open" and "free" quite frequently and interchangeably. Hence, looking at the arguments above, open-source software may be perceived as a roadmap to further innovation, which is constrained-free and open to all who want to collaborate thus creating a new business model.

BUSINESS MODEL

On the business model, open innovation is arguably perceived as having "inflows" and "outflows" of knowledge to innovate and to design a better and purposeful product.[23] Further, open innovation is a repository of abundant knowledge that must be readily used to create value and the knowledge must flow from one individual or the company to another individual(s) or company(s) or/and vice-versa. This explains how open innovation leverages external sources to save cost and time and hence produces better product at lower costs.[22] Thus, it can be seen that open-source software is an open innovation that brings disparate sources together to contribute, modify, and further distribute to remodify at a faster rate and at lower cost. Besides, open-source software can also be used and integrated with other products such as selling the support service with software, free basic versions versus value added advanced versions, integrating the free open software with other IT's infrastructure of the consumers.[22] For example, IBM is a good example of how it leveraged the open-source software landscape to boost its business model that lowered the costs of the operating system that IBM sells along with its own offerings that connected IBM's other hardware and software together.

On the contrary, it is also argued that open-source software should not only be seen as a source code that can mutate whichever way the society wants but should also be seen as a business from the legal and economic perspective. This explains why open-source software should be seen as free, that is public and nonproprietary, and requires modification, distribution with any binary attached and redistribution.[18] Hence, it becomes important to see its impact on the society as a whole and if it has any legal and economic implications. Using social construction of technology theory, open-source software product emanates as a result of constant pull and push between the society and the technology,[24] which arguably elaborates the curiosity of the open-source community and the way they code together as best as possible hence enjoying the freedom.[7] It becomes evident that it is the society, which shapes the product as per their needs and requirements. For example, RedHat, which sells a specific version of Linux, is one of the most famous and prevalent start-ups shaped around open-source technologies. It is based on customer service-support, consulting and quality verification. Another such example is Covalent, which is also a leader in relation to providing services and products for Apache, free and open-source software, characterized by consensus-based collaborative software development process. Apache

products are again based on full commercial service for the Apache Web server.[25] But the recent literature on open-source software also focuses on the paradox of why individuals should invest in producing such public goods in the absence of intellectual property rights that gives certain rights to individuals for its distribution and use. The resolution to this paradox lies in an explanation that intellectual property rights lie in the public domain that evolves a type of production model in which using and modifying the codes are accomplished through a dispersed and distributed community without paying a license fee, which has a bearing on innovation. Moreover, such innovative model of programming provides not only intrinsic rewards, but also additional extrinsic rewards to coders. An intrinsic reward refers to getting satisfaction through helping out in the form of "gift-giving" or by being altruistic as a responsible member of the community that sustain distinctiveness and reciprocity. The extrinsic award refers to the positive contribution of the programmer that will earn him reputation, thereby indicating his/her merit in the competitive job market.[25] Open-source software model also provides lessons to implement and structure innovation through self-selection, meritocracy, and peer-recognition that further ensures better product quality. Such mix of motives also encourages innovation and thus provides the programmers of open-source software an enhanced reputation for both monetary-based incentives and intrinsic-based incentives among peers and community identification.[26] Furthermore, the tenets of open-source software model motivates behavior that augments communication quality and cognitive trust that fosters identification with the project team, thereby leading the programmers to an affective trust that improves team effectiveness, which further reflects on better innovation, motivation, and enhanced reputation.[27] Hence, open-source software is characterized as follows.

- *A methodology for research and development*
- *A new business model (requiring new mechanisms for compensation and*
- *profit*
- *The 'defining nexus' of a community geared towards the development of*
- *common goods*
- *A new 'production structure' unique to 'knowledge economies'*
- *Even a political philosophy,"[18] pp. 180).*

Therefore, open source may be seen as having brought transformational changes as a result of the "pull" and "push" of social and economic landscape evolving through the lenses of "extraordinarily diverse hopes and fears" arising out of the "information revolution."

Open-source software model has come in conflict with two opposite extreme views—capitalist corporate and communism or socialism. This has generated immense tensions among the open-source model, hacker ethic, and political and economic ideologies, including capitalism, communism, and socialism. The literature suggests that a number of open-source proponents have argued that the success of free or open-source software movement has been because of its association with socialist and communistic ideologies. This has made capitalist corporate wary of deploying and implementing open-source principles as a primary method for developing and distributing open-source software, however, in recent years, such concerns have been considerably reduced and open-source model has been deployed in corporate computing environments.[28] As Perkins[29] claims that individual freedom is a thread that ties open-source software movement and capitalism. Furthermore, MacKenzie argues, "open source software cannot be separated from open source practices of socializing" (MacKenzie, 2001, as cited in the article by Jesiek.[28] Such arguments imply that open-source software movement seems to be lacking enough values to attain the vision that Feenberg promotes—a vision that leans more toward socialist ideals than toward individualism or capitalism.[30] Similarly, Himanen[17] also hints at historical tensions between hackers' values of freedom in open-source communities and capitalist enterprises. Such emphasis on individual freedom resonates with libertarian philosophies that have been very popular among technologists, which infer that hackers in the open-source community lean toward libertarian principles.[17] However, there is a paradox that seems to exist in the argument, "libertarians celebrate the cult of the individual but Open Source celebrates the collective. What does it mean to be an Open Source libertarian?" (Brate, 2000 in the article by Jesiek).[28] This paradox has been explained by Castells[31] pp. 33 as he contemplates the several uses of the term "libertarians": "'Libertarian'" has a different meaning in the European and in the American context. In Europe, it refers to a culture or ideology based on the uncompromising defense of individual freedom as the supreme value—often against the government, but sometimes with the help of the governments, as in the protection of privacy. In the U.S. context, "libertarian" is a political ideology that primarily means a systematic distrust of government, on the understanding that the market

takes care of everything by itself, and that individuals take care of themselves."

Now if one of the segments belonging to open-source proponents leans toward the former definition of libertarian and avoid the latter, the other segment may fervently challenge such leanings because they vociferously disdain the government, reify the open market and promote individuals.[32] Hence, open-source software movement has to inculcate more positive and deep democratic and social values in technologies as Feenberg[3] pp. 147 argues, "deep democratization promises an alternative to technocracy. Instead of popular agency appearing as an anomaly and an interference, it would be normalized and incorporated into the standard procedures of technical design." Therefore, we should tend to lean toward broader hacker community and beyond for values compatible with subversive rationalization and hence, societal good, and the business model of open-source software development can be discussed using Libertarian versus Corporate view.

DISCUSSION

Libertarian View Versus Corporate View

From the perspective of libertarians, the development of open-source software is perceived to be meant for societal good and a key to social progress based on "the culture of social gifts"[7] embedded in the domain of social and psychological perspectives, as envisaged by a collective model,[33] and not for profit and arising out of curiosity emanating as a result of collaboration among skilled parties located anywhere in the world.[7] Hence, we see that there is congruence between Raymond and Hippel and Krogh about how curiosity has given rise to some of the best software in the world. Moreover, through the examination of social movement theories, the use of free and open-source software has the ability to reduce the amount of profits obtained by commercial software to effect change to address social justice concerns in an attempt to develop a society that embraces the premises of fairness and impartiality for all.[8] Based on Feenberg's concept of philosophy of technology that technology cannot be disconnected from social values or goals, Hannemyr[34] compared and contrasted software developed under both software and commercial realms. His analysis provided further evidence about links between software codes and social values. He deduced that "Software constructed by hackers seem to favor such properties as flexibility, tailorability, modularity, and openness to facilitate on-going experimentation. Software originating in the mainstream is characterized by the promise of control, completeness, and immutability."[34] This forms a strong solidarity among the open-source developers based on sharing and coordination thereby, eliminating or reducing proprietary restrictions and increasing curiosity among the open-source coders that gives rise to the best software product in the market.[21] Furthermore, open-source community is not constrained by IP protection or monetary gain. As the open-source software development is contingent upon self-motivated attitudes of software developers,[35] researchers tried to explore what motivates the open-source community to develop free or open-source software.[36] They found intrinsic motivations associated with psychological factors, behavioral patterns, personal attitudes and attributes, and extrinsic motivations associated with direct compensation or job-related factors such as reputation, where reputation may be defined as a distribution of evaluations, opinions, or assessments about an individual, a group of individuals or an organization in an interest group having a specific concern or relationship with one another.[36]

Based on libertarian view, their motivation comes from self-improvement, learning, collaboration, and acknowledgment of their names and from the liberty to develop the software, whichever way the community wants.[21] However, Bitzer, Schrettl, and Schröder[37] argue that the motivation for open-source software development comes from altruism and the need for developing software for fun. Further, the person who is involved in developing open-source software is usually young and secures high value and benefits from the software in the form of "gifting" and "fun in coding" and faces a really low cost of development. Hence, open-source landscape represents a cultural and social solidarity among all the coders who are the users leading to strong bonding and cooperation.[38] Besides, "Libertarians see in open source a tool to emancipate individuals from governmental and corporate tyranny"[18] p. 180. On the contrary, it is argued, in "The Wealth of Networks," that not-for-profit research is more viable and economically efficient than for-profit research showing the role that nonproprietary production plays in shaping the system of information production, which leads the economists to advocate the possibility of government funding.[5] Moreover, the enthusiasts of the free market see open-source software as a product that is of high quality and innovative but it also creates a low barrier to entry in the market.[18]

As the use of open-source software Increased in popularity, efforts were made to normalize the process by associating the practice with freedom through the

identification of "free as in free speech, not free beer" as the term "open source software" was adopted to remove the associated political connotations.[38] As participation in the creation and use of open-source software increased, the quality of the software improved as flaws, or bugs, within the code were quickly identified and corrected. As open-source software relies on the involvement of numerous individuals to act as programmers, testers, and users, this participation for many became viewed as a method that allowed many software developers to learn and hone their craft.[5] From the managerial perspective, it was not just the learning, but also "recognition" and "reputation among one's peers" that contributors needed, which helped them get further work.[39] Lerner and Tirole[40] further argue that individuals participate in the development of open-source software because they want to show their ability of developing software for further career advancement. Seeing one's name on the open-source project was a big motivation for the contributors.[39] Hence, firms innovate and gain from "learning networks" of sharing and collaborating in the open-source landscape, which clearly shows the way Benkler tends to agree with Weber.

The corporate view perceives open-source software landscape as a business model that creates value[24] and hence competitive advantage. It is further argued that open-source software, which is a free software, when combined with other services or product, can bring profitability,[20] as envisaged by a private investment model,[33] and enhance market share. Further, Bessen[41]argues that firms use open-source software development as a way to increase their profits as they think that open-source software, as a public good, is more profitable to them than maintaining the software as a private code. Benkler clearly seems to be linking his view with this private investment model and argues that some of the coders, who participate in open-source software landscape may have the long-term financial motivation, such as service contracts or consulting etc.[5] and this, on the institutional level, can be explained by how IBM combined demand-side and supply side strategies in open source landscape and used Linux kernels to remarkably enhance its customer base for other products. This helped IBM generate US$ 2 billion in revenue. Therefore, open-source software, which is free software, is a business model for value creation when such open-source software is combined with other complementary services and products. Hence it may be termed as "Indirect Commercialization"[36] in the article by Ross.[36] This argument seems to be in agreement with Benkler. Hence, we see that the term "free" and "open" can not be used interchangeably and have different connotations as the open software matures.

Based on the libertarians' view, the motivation of open-source community comes from idealism, that is openness, and from freedom to use it and to distribute it,[39] whereas corporate view proposes their motivation comes from "reputation"[18] and "kinship amity" and "symbolic capital,"[7] which in the long run helps them get financial gains. Further based on the libertarians' view, free, or open software community emphasizes on social justice and solidarity, and hence, its structure of production is free from control and its distribution is free from "coercion of payment."[42] "Free art, and a free culture, is of vital importance for a free society" (Myers, 2008 in the article by Goncçalves and Figueiredo[42] pp. 215). The free culture movement enlarges its scope to include the idealistic goals of free or open software, "spreading freedom and cooperation"[43] pp. 129 to the entire culture and knowledge generation for a society that is creative and that is full of "sharism," where the more people collaborate or share, the better the product is and hence higher is the return (Mao, 2008 in the article by Goncçalves and Figueiredo[42]). This recognizes that "valuable" and "expensive knowledge" cannot be owned but is distributed equally for a societal good leading to emancipation, solidarity, and freedom.[44] Hence open-source software community is "a critique of existing laws, contracts, and business practices … [with] the potential to explicitly change the 'political-economic structure of society'" (Kelty, 2002 in the article by Chopra and Dex- ter[45] pp. xiv). Besides, coders do not rely on firm-based, market-based, or hybrid-based signals suggesting that it is the networked environments that generate the modality of organizational production creating "common based peer production" where "common" denotes the "institutional structure" permitting access to use, access, and control of resources.[5] It is this freedom that motivates coders to interact and collaborate among them without having any intervention from anybody without any fear, thus creating "commons" based efficiency for peer production. It is very important for efficient peer-production projects to be structured, both culturally and technically.[5] This motivates different users or individuals to code the program that is proportionate with the level of their motivation, ability, and availability. Alternatively, Benkler's arguments may also be seen in congruence with the corporate view that explains how IBM and Red Hat leveraged peer production to complement the business activities and hence attained competitive advantage in it, which clearly demarcates an interface between market and nonmarket actors

that may be seen as reinforcing each other.[5] This allows contributors to show their talent and debate on their ability to code resulting in their ego-satisfaction and self-expression, which eventually brings them "recognition" and "reputation"[48] that in the long term helps them get monetary benefits in the form of better job prospects. Hence ,corporate view examines the microeconomics of the contributors and their personal extrinsic motivations (Lerner & Tirole, 2001 in the article by Chopra and Dexter).[45] Research has found that motivation of open-source community comes from self-development through learning, receiving feedbacks, improving one's expertise and skills, signaling competence to possible employers, and gaining recognition from peers, which in turn are directly related to maintaining "reputation" leading to gaining better job opportunities.[35,36] They further argue that the companies looking for a particular set of skills can look into the open-source codes and can then determine whom to hire. Thus, gaining reputation becomes an important factor for open-source community, thereby helping them to advance their career prospects in the software industry.[46] Moreover, Rober *et al.*[35] found that paid participation, which was invariably linked to maintaining "reputation," encouraged the developers to contribute to the above average-level of open-source software development. Hence, this phenomenon may be argued to be a part of the new emerging economy incubating within the old one; thus leading to the development of "network economy," which is distributed and decentralized, integrating all across the globe and capable of bringing every human effort to full productivity that we can later address as consistent growth with a learning curve growing constantly, having better living standards and having lower level of unemployment.[47] Hence, it can be seen how open-source software landscape can be lever-aged in generating new economic structure and profitability.

From the perspective of the organization, libertarians perceive the open-source landscape as having a flat structure as opposed to hierarchical organization structure in the "proprietary" landscape.[5] Such a flat structure is arguably more beneficial and efficient in producing the quality of coding and has a positive effect on motivation.[21] By contributing more to open source, such coders can take control of the source repository.[38] However, the corporate view argues that free and open landscape is a platform for horizontal decision-making based on "charisma"—the ability to control and motivate contributors without formally having control over them.[48] This helps coders to contribute as much as they want without having to be controlled by anybody.

Besides, the emergence of open software landscape is arguably seen as deeply embedded in anti-capitalism, but yet the open-source software landscape is perceived as a new route to generating monetary value rooted in contemporary capitalism. However, coders do not just work on open-source software projects just because they want to do good to society but, based on the corporate view, they invest their time and effort in expectation of new form of capital "reputation,"[18] which will help them get their job prospects in future and hence financial gain eventually. The coders do not expect such gain immediately. There is an abundance to code in an open-source landscape and the coders are judged by how much they contribute in such landscape,[18] which clearly shows how open-source movement originated in the ethos of anticapitalism and antiauthoritarianism having the elements of socialism or "socialized process" embedded in the continued development of "socio-technological base"[36] in the form of "solidarity," "sharing," and "collaboration" for the societal good and then transformed itself into "collaborative capitalism"[48] bringing monetary value to all the members of the society, who contribute to open-source landscape.

Libertarians justify and view free riders as important contributors in an open-source landscape, although they may not be qualified enough to code with quality. From the economics, managerial, and technical perspective, free riders are of almost no value but they are likely to find errors in the software for advanced coders,[7] thus giving a specific direction to the open-source project. Further, Johnson[49] analyzed the cost and benefit of developing the open-source software versus not developing it based on game theory model and concluded that the open-source programmers will be motivated to develop code in collusion with free riding as Agrawal12 argues that free riding should not be ignored as it prevents waste and make the programming more efficient, which in turn motivates the open-source programmers to code even further. This helps them gain learning for further contribution. However, the corporate view argues that free-riders are nonentities and they do not pay any attention to them since many successful companies are getting inclined to use open-source software for their business-critical solutions.[50]

Finally, when we juxtapose the libertarian and corporate view, the model becomes private collective model,[33] which proposes that open-source software is not just for "public good" but it also has the elements of "private" or financial gain and offers "best of both worlds." On the one hand, it becomes

a source of innovation and brings solidarity and learning curves as a societal good. On the other hand, it provides opportunities and "selective incentives" to its contributors.

CONCLUSION

Open source software has become a very vibrant and competitive methodology and a business model for further research and development. For example, certain offerings, such as Linux and Mozilla, challenged the monopoly on personal computing by offering options to the operating systems and web browsers presented by Microsoft and Apple. Even though the major corporations have the financial backing to provide programmers and software engineers with compensation for their efforts and contributions, open-source enthusiasts known as hackers, continue to challenge these accepted practices by developing software without expectation of compensation.

The review shows how open-source software development has evolved over time and has gradually ingrained itself in the social, cultural, and economic landscape of the society that are inseparable from each other and hence, has been a cause of freedom, social justice, and emancipation from control. Further, the review seems to have agreed with Feenberg's argument about critical theory of technology or/and philosophy of technology[3] that "the issue is not simply society's responsibility for controlling technology, but extends to a reflexive transformation of technical disciplines themselves as the design process becomes socially conscious"[3] pp. 91. Moreover, the review also explores the link between personal values and motivations that help explain how open-source software developers get motivated and how open-source software projects recruit, train, and cultivate in them with a high level of expertise so that they can become efficient contributors. Finally, the review of the literature above, through the analysis of both libertarian and corporate view, shows that open-source software development model is not just based on "societal good" but also concurrently based on a "business model," which is linked to "monetary gain" and hence both the terms—"societal good" or "freedom" and "business model" or "benefit"—are not "mutually exclusive" but complement one another.

The review also found some research gaps about how social media can be used to help proliferate the development of open-source software. Hence, further research may be conducted on how and, to what extent, social media networks can be used that can contribute to the development of open-source software community and how such social networks can enhance the reliability and monetary profitability of the open-source software development.

ACKNOWLEDGMENTS

There was no funding required for this research project.

REFERENCES

1. D. Howe, "Open source," *Free Online Dictionary Comput.* 29 December, 1999. Accessed on: August 5, 2019. Retrieved from http://foldoc.doc.ic.ac.uk/ foldoc/ foldoc.cgi? query¼openÞsource
2. T. Bollinger, "Use of free and open-source software (FOSS) in the U.S. Department of Defense," 2003. Accessed on: October 26, 2019. Retrieved from https:// www.researchgate.net/ publication/281562475_Use_ of_free_and_opensource_ software_FOSS_in_the_US_ Department_of_Defense
3. A. Feenberg, *Questioning Technology*. London, U.K.: Routledge, 1999.
4. J. Webster and R. T. Watson, "Analyzing the past to prepare for the future: Writing a literature review," *MIS Quar.*, vol. 26, no. 2, pp. xii–xxiii, 2002. Accessed on: November 3, 2019. Retrieved from https://web.njit.edu/
5. Y. Benkler, *The Wealth of Networks: How Social Production Transforms Market & Freedom*. New Haven, CT, USA: Yale Univ. Press. pp. 35–383, 2006. Accessed on: December 4, 2018. Retrieved from http://dlc.dlib.indiana.edu/ dlc/bitstream/handle/10535/7396/Wealth-of-Networks.pdf;jsessionid¼2EDDC653F33676B11B4B665AAEF2E918?sequence¼1
6. N. Tkacz, "From open source to open government: A critique of open politics," *Ephemera: Theory Politics Org.*, vol. 12, no. 4. pp. 386–405, 2012, Accessed on: Aug. 15, 2020. Retrieved from http:// wrap.warwick.ac.uk/ 53295/1/WRAP_Tkacz_12-4tkacz.pdf
7. E. Raymond. *The Cathedral and the Bazaar: Musings on Linux and Open Source by an accidental revolutionary*. Sebastopol, CA, USA: O' Reilly and Associates, 2001.
8. J. L. Sullivan, "Free, open source software advocacy as a social justice movement: The expression of F/OSS movement discourse in the 21 century," *J. Inf. Technol. Politics*, vol. 8, no. 3, pp. 223–239, 2011.
9. S. Chopra and S. Dexter, "Free software and the economics of information justice," *Ethics Inf. Technol.*, vol. 13, no. 3, pp. 173–184, 2011.
10. M. O. Arslan, "Free and open source software as a public good: Implications for education," *Eur. J. Res. Educ.*, vol. 2, pp. 158–165, 2014.

11. R. Rajala, M. Westerlund, and K. M€oller, "Strategic flexibility in open innovation – Designing business models for open source software," *Eur. J. Marketing*, vol. 46, no. 10, pp. 1368–1388, 2012..
12. D. R. Agrawal, "If open source code is a public good, why does private provision work (Or does It)?" *LBJ J. Public Affairs*, vol. 18, no. 2, pp. 55–64, 2006.
13. E. G. Coleman, *Coding Freedom: The Ethics and Aesthetics of Hacking*. Princeton, NJ, USA: Princeton Univ. Press, 2013.
14. C. Kelty, *Two Bits: The Cultural Significance of Free Software*. Durham, NC, USA: Duke Univ. Press, 2008.
15. E. Raymond, *The New Hackers Dictionary*. Cambridge, MA, USA: MIT Press, 1996.
16. G. Moody, *Rebel Code: The Inside Story of Linux and the Open Source Revolution*. Cambridge, MA, USA: Perseus, 2001.
17. Himanen. *The Hacker Ethic and the Spirit of the Information Age*. New York, NY, USA: Random House, 2001.
18. S. Weber, "The political economy of open source software and why it matters," in *Digital Formations: IT and New Architectures in the Global Realm*, R. Latham and S. Sassen Eds., Princeton, NJ, USA: Princeton Univ. Press, 2005, pp. 178–212.
19. D. Bretthauer, "Open source software: A history," *Inf. Technol. Libraries*, vol. 21, no. 1, pp. 3–10, 2001. Accessed on: December 4, 2018. Retrieved from https:// opencommons.uconn.edu/cgi/viewcontent.cgi?article¼1009&context¼libr_pubs
20. B. Fitzgerald, "The transformation of open source software," *MIS Quart.*, vol. 30, no. 3, pp. 587–598, 2006.
21. E. V. Hippel and G. V. Krogh, "Special issue on open source software development," *Res. Policy*, vol. 32, no. 7, pp. 1149–1157, 2003. doi: https://doi.org/ 10.1016/ S0048-7333(03)00054-4
22. H. Chesbrough, "The era of open innovation," *MIT Sloan Manage. Rev.*, vol. 44, no. 3, pp. 35–41. 2003. Accessed on: December 5, 2018. Retrieved from http://www. humanitarianinnovation. com/ uploads/7/3/4/7/ 7347321/chesbrough_2003.pdf.
23. L. T. R Torres, E. R. B. Ibarra, and A. P. L. Arenas, "Open innovation: A literature review of case studies," *J. Adv. Manage. Sci.*, vol. 3, no. 4, pp. 362–365, 2015.
24. A.J. Flanagin, C. Flanagin, and J. Flanagin, "Technical code and the social construction of internet," *New Media Soc.*, vol. 12, no. 2, pp. 179–196, 2010.
25. B. Kogut and A. Metiu, "Open source software development and distributed innovation," *Oxford Rev. Econ. Policy*, vol. 17, no. 2, pp. 248–264, 2001.
26. G. von Krogh, "Open source software development," *MIT Sloan Manage. Rev.*, 2003. Accessed on: Oct. 27, 2019. Retrieved from https:// sloanreview.mit.edu/ article/opensource-software-development/
27. K. J. Stewart and S. Gosain, "The impact of ideology on effectiveness in open source software development teams," *MIS Quar.*, vol. 30, no. 2, pp. 291–314, 2006.
28. B. K. Jesiek, "Democratizing software: Open source, the 1076 hacker ethic, and beyond," *First Monday*, vol. 8, no. 10, 2003, Art. no. 1077. Accessed: October 30, 2019. [Online]. Available: https:// firstmonday.org/article/ view/1082/1002#j4
29. G. Perkins, "Culture clash and the road to world domination," *IEEE Softw.*, vol. 16, no. 1, pp. 80–84, Jan./ Feb. 1999.
30. G. Kirkpatrick, "Transforming dystopia with democracy: The technical code and the critical theory of technology," in *Critical Theory and the Thought of Andrew Feenberg*. D. P. Arnold and A. Michel, Eds. New York, NY, USA: Palgrave Macmillan, 2017, pp. 263–282.
31. M. Castells. *The Internet Galaxy: Reflections on the Internet, Business, and Society*. Oxford, U.K.: Oxford Univ. Press, 2001.
32. R. Barbrook, "Cyber-communism: How the americans are superseding capitalism in cyberspace," *Sci. Culture*, vol. 9, no. 1, pp. 5–40, 2000.
33. E. V. Hippel and G. V. Krogh, "Open source software and the 'private collective' innovation model: Issues for organization science," *Org. Sci.*, vol. 14, no. 2, pp. 209–223, 2003.
34. G. Hannemyr, "Technology and pleasure: Considering hacking constructive," *First Monday*, vol. 4, no. 2, 1999. Accessed: November 5, 2019. [Online]. Available: http:// www.firstmonday.org/issues/issue4_2/gisle/
35. J. Roberts, I.-H. Hann, and S. Slaughter, "Understanding the motivations, participation, and performance of open source software developers: A longitudinal study of the apache projects," *Manage. Sci.*, vol. 52, no. 7, pp. 984–999, 2006.
36. S. Krishnamurthy, "On the intrinsic and extrinsic motivation of free/libre/open source (FLOSS) developers," *Knowl. Technol. Policy*, vol. 18, no. 4, pp. 17–39, 2006.
37. J. Bitzer, W. Schrettl, and P. J. H. Schr€oder, "Intrinsic motivation in open source software development," *Free Univ. Berlin Discussion* Paper No. 2004/19. Accessed on: October 31, 2019. Retrieved from https://www.econstor.eu/bitstream/10419/49926/1/668836490.pdf
38. R. Stallman, "Viewpoint: Why 'Open Source' Misses the point of free software," *Commun. ACM*, vol. 52, no. 6, pp. 31–33, 2009.
39. Y. Li, C. H. Tan, and H. H. Teo, "Leadership characteristics and developers' motivation in open source software development," *Inf. Manage.*, vol. 49, no. 5, pp. 257–267, 2012.
40. J. Lerner and J. Tirole, "Some simple economics of open source," *J. Ind. Econ.*, vol. 50, no. 2, pp. 197–234, 2003.

41. J. Bessen, "Open source software: Free provision of complex public goods. Research on Innovation Discussion Paper, Jul. 2005. Accessed on: October 31, 2019. Retrieved from http://www.researchoninnovation.org/opensrc.pdf
42. N. Goncçalves and M. Figueiredo, "OpenLab ESEV – A narrative of libre software and free culture in a Portuguese higher education institution," in P. Missomelius, W. S, T. Hug, P. Grell, and R. Kammerl, Eds. *Medien-Wissen-Bildung: Freie Bildungsmedien Und Digitale Archive*. Marburg, Germany: Innsbruck Univ. Press, 2014, pp. 211–225.
43. R. Stallman, "Copyleft: Pragmatic idealism," in *Free Software, Free Society: Selected Essays of Richard Stallman*, 2nd ed. J. Gay Ed. Boston, MA, USA: Free Softw. Found., 2010, pp. 129–134.
44. C. Doctorow, "You can't own knowledge," in *Freesouls: Captured and Released*. J. Ito Ed. Taiwan: Adams, 2008, pp. 108–110.
45. S. Chopra and S. Dexter, *Decoding Liberation: The Promise of Free and Open Source Software*. London, U.K.: Routledge, 2008.
46. K. Lakhani and E. von Hippel, "How open source software works: Free user-to-user assistance," *Res. Policy*, vol. 32, no. 6, pp. 923–943, 2003.
47. M. Castells, "The information society, the new economy and the network society," in *The Information Society Reader*, F. Webster, Ed. London, U.K.: Routledge, 2004, pp. 150–164.
48. A. Barron, "Free software production as critical social practice," *Econ. Soc.*, vol. 42, no. 4, pp. 597–625, 2013.
49. P. Johnson, "Open source software: Private provision of a public good," *J. Econ. Manage. Strategy*, vol. 11, no. 4, pp. 637–662, 2002.
50. P. Finnegan and L. Morgan, "Beyond free software: An exploration of the business value of strategic open source," *J. Strategic Inf. Syst.*, vol. 23, no. 3, pp. 226–238, 2014.

RIZWAN AHMAD research interests include information system management, digital surveillance and privacy, cybersecurity, management of open source software development, and data privacy. He has been associated with various research projects. He has also been associated in carrying out research activities with the Department of Management, and the Department of Media and Communication, London School of Economics & Political Science in London. He received the Master's degree in technology management from the School of Professional Studies, Columbia University in the City of New York, New York, NY, USA. He is currently working toward the Ph.D. degree with the University of Sydney Business School, Darlington NSW, Australia. Contact him at: ra2770@columbia.edu