EFFORTS TO SUCCESSFULLY ADD ETHICS TO CS EDUCATION

A Research Paper submitted to the Department of Engineering and Society

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Matt Koehler

Spring 2023

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Advisor

S. Travis Elliott, Department of Engineering and Society

INTRODUCTION

Computer Science (CS) is a field that is constantly going through rapid growth and change. With this rapid growth, the impact CS makes on everyday life has also grown. People are using the internet on average 7 hours a day at this point, and 96.6% of internet users aged 16-64 own a mobile phone (Kemp, 2022). So it can be seen with the obvious impacts like iPhones and social media; however, software can influence everyday life more than just that because so much of today is powered through software. With the field of computer science becoming so much more impactful, the ethical implications in the field of CS are becoming increasingly more important, and thus the education of CS must also become better as well. This is because if new software engineers are to be wielding such power in this field they must be educated in ethics to ensure that the software they are creating does not negatively impact society. Overall, the goal of this study is to look at the social factors of ethics education in computer science and this will be done using the social construction of technology (SCOT) framework. Ultimately, the ethics curriculum in this case will be the technology that is influenced and shaped by relevant social groups such as CS professors, colleges, CS students, current software engineers and policy makers. These are the social groups that have and will continue to influence ethics curriculum in computer science.

CURRENT PROBLEMS

Ethics in software engineering has always been a pressing topic because of how much software affects society. For example, there are many ethical dilemmas facing software engineers today such as privacy, algorithmic bias, and data collection (Sweeney, 2022). Furthermore, even if a person makes a great app it runs the danger of the app becoming too addictive which can

lead to disastrous consequences such as a lot of the problems with social media (Lawton, 2020). So, it is important to get to the root of this beginning with the education of software engineers on ethics (Narayanan & Vallor, 2014, p. 23). Gülcan in particular emphasizes the importance of ethics and its education as education is more than just concepts but becoming a better human (2014, 2625). It is imperative that future software engineers are taught ethics and while many computer science programs have classes for ethics it is necessary to find the best way to implement ethics into computer science education.

There is an ongoing conversation on how exactly to implement ethics in schools; for example, The University of Virginia which has classes just for ethics on the side and not included within the core CS curriculum (University of Virginia, 2022). This is done in the form of Science, Technology and Society (STS) classes which have the goal of teaching ethics to all engineers by educating students on the relationship of technology and society. However, Melo and de Sousa (2017) are critical of this and say ethics needs to be implemented within the computer science classes themselves (p. 45). A new curriculum made a suggestion similar to this as it said to implement ethics within the core computer science classes (Hadfield et al., 2019). This is seen in Figure 1, where there are many differing ideas on how to implement ethics into the computer science curriculum and whether to just have classes on the side or have the core CS classes themselves have ethics within them. This subject needs further research as there is no consensus on what method of teaching ethics is the most effective for students. It is imperative to learn a better method for the teaching of ethics in CS as it will lead to future software engineers being ethical in their futures. This has a massive impact as it will lead to more ethical technologies being created than what were created previously.

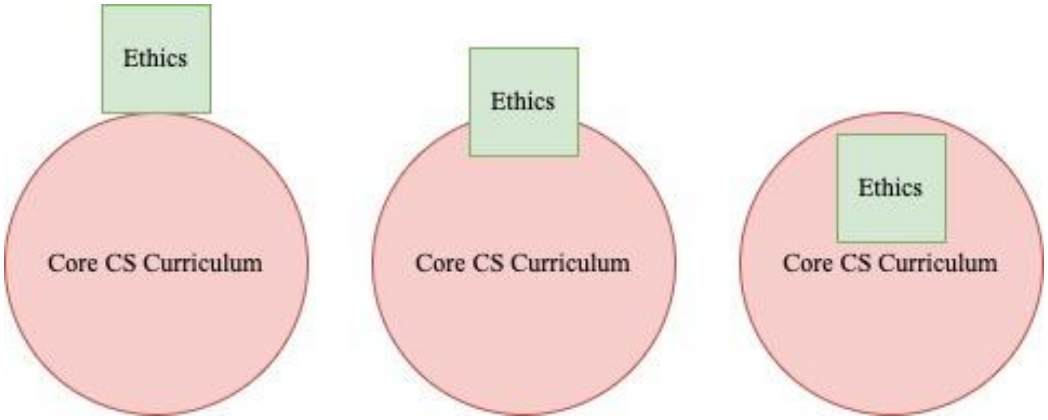

Figure 1: Differing Ideas On Ethics In CS Curriculum. The figure illustrates the differing ideas on how to implement ethics into the core computer science curriculum. This is seen as the circle on the left is completely separate from ethics while the circle on the right it is shown that ethics and core CS curriculum are to be taught together. (Koehler 2022).

SOCIAL CONSTRUCTION OF TECHNOLOGY

Social construction of technology is the STS framework that will be used to analyze this problem. SCOT states that human action is what shapes technology not the other way around which is what the theory of technological determinism states. An important concept for SCOT is interpretative flexibility which means that each technology has a different interpretation for various groups. The relevant social groups to look at in the case of teaching ethics to computer science students would then be the CS professors, colleges, CS students, current software engineers and policy makers. These different groups all shape how exactly ethics is taught to computer science students and have different ways they interact with this teaching of ethics from being a producer of this like professors or being a user like the students. Also, some groups such as software engineers are not directly users or producers of the technology but still have a massive effect on how exactly ethics will be taught to computer science students. A core concept of interpretive flexibility is the problems and conflicts that come from the relevant social groups. This is very clear in the case of teaching ethics because each social group has certain preferences

on how ethics should be taught, and this conflict needs to be resolved someway. SCOT also has another important concept such as closure which can apply here as ethics in computer science as it is very likely that as these conflicts between the social groups resolve a new redefinition of the problem of ethics curriculum can arise. Overall, SCOT looks at how different groups can shape technology, and in this paper SCOT will specifically be used to look at how various social groups shape how ethics is taught to computer science students.

PROFESSORS

The first social group to look at are professors as they have a profound impact on ethics learning of students. The professors can be seen as the producers of the technology in this case as they produce the ethics curriculum for students to consume. They shape ethics by generally creating the curriculum in their classes; however, how exactly computer science ethics is implemented can vary for many reasons. For example, Hadfield et al. are professors that have proposed a curriculum that implements ethics within core computer science classes themselves (2019). This would mean CS professors would have to be the professors to teach ethics to CS students, but this would be a hard adjustment for CS professors because they generally have expertise in computer science, but they do not have expertise in ethics and generally leave this teaching for ethics professors. This happens at The University of Virginia where the usage of STS (science, technology and society) classes are used to teach engineering students ethics (University of Virginia, 2022). While this could be beneficial because the teaching of ethics by ethics professors is very likely to be much better than a CS professor, this type of approach does lack the ability to break down ethics cases that have specific computer science implications. This is an important point and is why many professors have proposed, such as Melo and de Sousa,

that ethics must be included within the general CS curriculum a student takes because these specific cases must be covered (2017, p. 45). This emphasizes interpretive flexibility as professors in this instance are able to shape how students consume ethics learning, but they do not hold all the power as there are other social groups that also shape computer science ethics learning. Overall, professors as stakeholders want to educate their students the best they can, which is their ultimate goal, but clearly they have differing opinions on how exactly best to educate their students. As the producers of ethics curriculum, professors have immense power in shaping ethics learning in computer science.

COLLEGES

Colleges are another relevant group that can influence computer science ethics curriculum. This is because colleges are ultimately the ones responsible to employ professors and make the decision on what exactly it takes for certain majors to graduate. For example, as discussed before The University of Virginia has separate classes for ethics learning for their computer science students. However, at The University of Southern California (USC) there are not any of these additional ethics classes; therefore, all ethics learning for CS students would have to be done during their CS classes (University of Southern California, 2023). This points out one of the key components of interpretative flexibility which creates the problems and conflicts between social groups. This is because while many CS professors may want to educate their students in ethics, some colleges can take this away by adding separate ethics classes which should do the bulk of ethics learning. Since colleges can choose what classes a student must take to graduate, they can have an immense impact on how CS students learn ethics. Furthermore, colleges must make the tough decision on what exact classes a student should be required to take

in order to graduate and whether to focus more on a student's technical skills or ethical skills. This tradeoff is a decision these colleges make emphasizing their role in shaping how ethics is taught to computer science students because some colleges like UVA may choose to have ethics classes that are required for all computer science majors while colleges like USC choose to focus on a student's technical skills.

Additionally, colleges have to grapple with the fact that ethics classes do not really work in teaching students ethics. This was shown in a study about a business ethics class where they found “that taking a course in ethics did not significantly affect student attitudes” (Peppas & Diskin, 2001). While this was a business ethics class, it is quite likely that the attitudes of computer science students could very well be the same. Colleges then must look at whether ethics classes are even worth it in the first place or look at how to better implement ethics classes into the curriculum in order to actively change student’s attitudes on ethics. A college’s outlook on these issues has a massive effect on how ethical curriculum is enacted and thus shaping the technology of ethics learning.

STUDENTS

Students are another relevant social group that shapes the computer science ethics curriculum. Students in this case are the users of the ethics curriculum which would mean they would want the best “product” possible. Students also have a lot of agency with choice in this case as they can mostly choose everything from which college they attend to which professor they take a class with. This allows each student to have agency on shaping their own ethical learning. However, students have an even greater impact on ethics curriculum as many college students themselves are not ethical (Boudreau, 2022). With students not being ethical, it is very

difficult to get them to learn ethics which goes back to one of the core concepts of interpretive flexibility which states that there are conflicts between groups that are hard to resolve and there need to be solutions to these conflicts. The conflict here is that students are being taught to be ethical while not being ethical themselves. However, there are some solutions to this such as putting ethics at the forefront of discussion and encouraging ethical decision making (Boudreau, 2022). Academic integrity is important and can be accomplished through solidarity which can make students have an appreciation for ethics and thus be open to learning ethics (Bieliauskaitė, 2021).

In a study of computer science students in a CS ethics class it was found that "despite emphasizing ethical theories and frameworks in the course, most students did not use these theories in their decision-making process" (Hedayati-Mehdiabadi, 2022). The study exhibits a huge problem because as users of the "product" of ethical learning they are not utilizing the "product" to its fullest potential. This problem shows how students affect ethical learning because by not fully learning ethics this will cause future ethical problems which then will cause computer science ethics learning to change once again to adapt. Furthermore, if students are not being able to apply key concepts that are taught to them that would suggest the curriculum would need to change. Overall, computer science students and their actions shape how computer science ethics learning takes place.

SOFTWARE ENGINEERS

Software engineers also have an impact on computer science ethical education. This is because they create the ethical problems that must be addressed within computer science education. These ethical problems such as privacy data collection and algorithmic bias are just a

few dilemmas that software engineers face daily (Sweeney, 2022). Software engineers can routinely fail at these ethical decisions thus making it more important to not repeat these mistakes. In order to not repeat these mistakes it is important to teach the ethics to future software engineers so ultimately they do not make the same mistakes past software engineers made. Software engineers affect how ethics is taught because as they grow the field new software will be created with new ethical problems. For example, ethics of artificial intelligence (AI) is a new problem that has not been explored much as AI is relatively new with ChatGPT being the most recent innovation from the field of AI. AI can create malware, deep fakes and phishing attacks so it creates a new ethical problem of how to distribute this new technology of AI (Air et al., 2023). Another example is social media and the ethics of privacy and data collection (Vallor, 2021). These are both newer problems that have to now be taught to computer science students showing the impact that current software engineers have on ethics learning for CS students. These ethical problems will continue to grow as time goes on as software becomes more advanced which means ethics learning will have to grow as well. While current software engineers do not directly produce ethics curriculum or learn ethics curriculum, they have a massive impact on the ethical curriculum of computer science students.

POLICY MAKERS

Policy makers also have an influence on computer science ethical learning. This is because even though they are not directly involved within the making of ethics curriculum they make laws that software engineers must follow. Following laws could also be considered ethical if using a deontological view; therefore, CS students learning laws specific to their field is necessary to get a better understanding of ethics. Laws have already been passed such as social

media privacy laws which is a big ethical dilemma within software engineering currently (Thomson Reuters, 2018). The laws outline the obligations of a social media company which can have an impact on ethical learning as these laws can be used as a baseline for ethics within computer science. This is further seen with cryptocurrency which is a relatively new technology that has many new laws being made to regulate it (Tarkowski & Appleton, 2023). This displays that policy makers will always make laws to regulate any new technology thus showing their impact in trying to impart some type of ethics onto these new technologies and how they are used. This trickles down to ethical learning as CS students should learn about these new regulations as many laws were made for a specific ethical reason. Ultimately, policy makers can create laws that influence current ethics within software engineering which should then be passed onto computer science ethical curriculum.

CLOSURE

Closure is the second stage of SCOT and it can be achieved in two ways. One is rhetorical closure in which the social groups see the problem being solved and thus the need for change is finished. Rhetorical closure however is not likely to be applicable in the situation of CS ethics education as the field of CS is ever changing so the conflicts between the social groups will likely never be solved. The second way to achieve closure is redefinition of the problem. This on the other hand could occur as the field of computer science changes over time there may be a massive change in the ethics of it, which can then create new conflicts for the relevant social groups that would need to be resolved. This would redefine what exactly the problem is thus bringing these current specific conflicts between the relevant social groups to closure.

CONCLUSION

There are many social groups that impact how ethics is taught to computer science students. All of the social groups previously discussed and probably many more ultimately shape how ethics has been taught and will be taught in the future to CS students. The takeaway from this is that to teach ethics successfully it is impossible to just look at one particular problem and fix that problem. It would be impossible to just fix the ethics curriculum and expect ethics to be perfectly taught to all CS students because all of these groups influence that curriculum and how exactly ethics will be taught. It is important to focus on every relevant social group in order to break down why exactly ethics is being taught the way it is. Once it is seen why computer science ethics teaching is lacking then it will be much easier to come up with a solution that can work for all relevant stakeholders.

References

Air, C., Wijetunge, S., & Dimitrov, A. (2023, February 28). The ethics of AI: The cyber risks posed by Chat GPT. DAC Beachcroft. Retrieved March 17, 2023, from https://www.dacbeachcroft.com/en/articles/2023/february/the-ethics-of-ai-the-cyber-risks-posed-by-chat-gpt/

Bieliauskaitė, J. (2021). Solidarity in academia and its relationship to academic integrity. Journal of Academic Ethics, 19(3), 309–322. https://doi.org/10.1007/s10805-021-09420-6

Boudreau, E. (2022, April 5). The questionable ethics of college students. Harvard Graduate School of Education. Retrieved March 17, 2023, from https://www.gse.harvard.edu/news/uk/22/04/questionable-ethics-college-students

Gülcan, N. Y. (2015). Discussing the importance of teaching ethics in Education. Procedia - Social and Behavioral Sciences, 174, 2622–2625. https://doi.org/10.1016/j.sbspro.2015.01.942

Hadfield, S., Weingart, T., Coffman, J., Caswell, D., Fagin, B., Sarmiento, T., & Graham, P. (2019). Streamlining Computer Science Curriculum Development and assessment using the new ABET student outcomes. Proceedings of the Western Canadian Conference on Computing Education. https://doi.org/10.1145/3314994.3325079

Hedayati-Mehdiabadi, A. (2022). How do computer science students make decisions in ethical situations? implications for teaching computing ethics based on a grounded theory study. ACM Transactions on Computing Education, 22(3), 1–24. https://doi.org/10.1145/3483841

Kemp, S. (2022, May 4). Digital 2022: Time spent using connected tech continues to rise - datareportal – global digital insights. DataReportal. Retrieved March 17, 2023, from https://datareportal.com/reports/digital-2022-time-spent-with-connected-tech

Lawton, G. (2020, December 22). 5 examples of ethical issues in software development. SearchSoftwareQuality. Retrieved December 1, 2022, from https://www.techtarget.com/searchsoftwarequality/tip/5-examples-of-ethical-issues-in-software-development

Melo, C. D., & de Sousa, T. C. (2017). Reflections on cyberethics education for Millennial Software Engineers. 2017 IEEE/ACM 1st International Workshop on Software Engineering Curricula for Millennials (SECM). https://doi.org/10.1109/secm.2017.10

Narayanan, A., & Vallor, S. (2014). Why software engineering courses should include ethics coverage. Communications of the ACM, 57(3), 23–25. https://doi.org/10.1145/2566966

Peppas, S. C., & Diskin, B. A. (2001). College courses in ethics: Do they really make a difference? International Journal of Educational Management, 15(7), 347–353. https://doi.org/10.1108/09513540110407947

Sweeney, M. (2022, February 11). Ethical dilemmas in computer science. ZDNET. Retrieved October 27, 2022, from https://www.zdnet.com/education/computers-tech/ethical-dilemmas-computer-science/

Tarkowski, D., & Appleton, R. (2023, March 13). State by state cryptocurrency laws and regulations: Bloomberg law. Bloomberg Law. Retrieved March 17, 2023, from https://pro.bloomberglaw.com/brief/cryptocurrency-laws-and-regulations-by-state/

Thomson Reuters. (2018, August 21). Social Media Privacy Laws. Findlaw. Retrieved March 17, 2023, from https://www.findlaw.com/consumer/online-scams/social-media-privacy-laws.html

University of Southern California. (2023). Academics. Program: Computer Science (BS) - University of Southern California - Acalog ACMS™. Retrieved March 17, 2023, from https://catalogue.usc.edu/preview_program.php?catoid=8&poid=7761

University of Virginia. (2022, March 31). Computer Science Undergraduate Advising. University of Virginia School of Engineering and Applied Science. Retrieved September 22, 2022, from https://engineering.virginia.edu/departments/computer-science/computer-science-undergraduate-advising#accordion656360

Vallor, S. (2021, August 30). Social Networking and Ethics. Stanford Encyclopedia of Philosophy. Retrieved March 17, 2023, from https://plato.stanford.edu/entries/ethics-social-networking/#SocNetSerPri