33.

Computational Fluency as Argumentation Support at the Community Level in Scratch

Crystle Martin (University of California, Irvine)

Abstract

In this paper I explore findings from an ethnography of the online Scratch community. Through the observations of the Scratch forums, I propose that aspects of computational thinking, which has been studied previously in Scratch, are used by Scratchers (i.e., people who participate on Scratch) as a way to strengthen arguments for topics about which Scratchers feel passionately. The analysis takes a social argumentation approach, which emphasizes context. The focus on context is important for a community like Scratch where the participants of the community feel strong connection and ownership over the activities and structure of community. This paper demonstrates at a community level how aspects of computational thinking are used to strengthen arguments, and what this does for Scratchers who use it.

Introduction

This paper focuses on an ethnography for the online coding community, Scratch. Scratch is a free, online multimedia authoring tool with over ten million registered members and over 13 million projects (Scratch, n.d.). Scratch, a visual coding language, was originally designed for middle school students but the online community currently has users that are younger than 8 and older than 70, with the majority of Scratchers between 9 and 16 years old. The volume of conversation in this community is equally large, with participants communicating in the studios, where Scratchers (i.e., people who participate on Scratch) curate projects; commenting on projects; and participating on the forums. At the time of writing, there are nearly 71 million comments across the site (Scratch, n.d.). Scratch supports a variety of learning related activities (Fields, Giang, & Kafai, 2013; Koh, 2013; Burke & Kafai, 2012), but it strongly supports the development of computational thinking (Brennan & Resnick, 2012). Computational thinking permeates the activities of the Scratchers, whether commenting on a project or arguing their position on the forum. As described by the Center for Computational Thinking (n.d.) at Carnegie Mellon, “Computational thinking is a way of solving problems, designing systems, and understanding human behavior that draws on concepts fundamental to computer science. To flourish in today’s world, computational thinking has to be a fundamental part of the way people think and understand the world.” This paper explores how computational thinking is used in argumentation in the online Scratch community.

Argumentation and Computational Fluency

Discussion and debate is common in online communities, especially those that have passionate participants, access to large amounts of information pertaining to the topic of the community, and that experience change and iteration. Argumentation has deep roots in research. One of the most prevalent frameworks in argumentation is Toulmin's model. Toulmin (1958) identified six parts that a strong argument should include: claim, data, warrants, rebuttals, backings, and qualifiers. This structured view of argumentation was developed long before the internet and takes a one side winning over the other approach, over a more consensus building view, which could be found in a collaborative online community. Toulmin's model was also created to study argumentation in real time and usually between a limited number of piece, unlike online communities.

Scholars are pushing back against this structured approach. They posit that individuals should build on the collective knowledge of the group, instead of taking a more adversarial approach of convincing others of their own viewpoint using argumentation (Andriessen et al., 2003a, b), and approaching argumentation from a constructivist approach (Leitão, 2000). Leitão (2000) contends that knowledge development is an important part of argumentation and should play a prominent role in analysis. She also criticized work like that of Toulmin's for not accounting for how participants' arguments transform over the course of a discussion.

Social argumentation in online communities functions differently than that of face-to-face situations, because of the recorded nature of the medium, the amount of time an argument can cover, and the number of individuals that can participate, which is a much greater number than can conceivably participate in a face-to-face argument. Social argumentation in general relies heavily on context (Walton, 1996). It is not enough to analyze a single excerpt of an argument, instead the type and goal of the argument must also be considered in order to have adequate context for analysis. De Moor and Efimova (2004) demonstrate how interest-driven spaces, as a context, promote argumentation, as well as learning (Martin, 2014; Steinkuehler, 2007). In the online game community of *World of Warcraft*, Alagoz (2013) found that the youth engaged in quality argumentation in a majority of their exchanges. She creates a streamlined approach with four elements: argument, counter-alternative, counter-critique, and other. Argument includes claim and evidence. Counter-alternative includes counter-claim and alternative evidence, which weakens an opponent's position by introducing new criticism rather than attacking the opponent's argument. Counter-critique includes counter-claim and refuting evidence, which is a more skilled attack that directly addresses an opponent's argument. Alagoz uses Other to encompass agreement, disagreement, submission, restatement, etc.

Creating a sound argument is difficult for both individuals and groups to master (Rourke & Kanuka, 2007). In certain situations, especially where the goal is knowledge growth as much as winning a point (Andriessen et al. 2003a, b), strong argumentation in online communities can support and be supported by other types of problem-based learning like computational thinking. Brennan and Resnick (2012) created a computational framework (Table 1) that breaks down into three main parts: computational concepts, computational practices, and computational perspectives. What is demonstrated in Table 1 is that computational thinking is problem-based, contextual learning, which can be developed in interest-driven spaces. Computational thinking is knowledge building, and in fact could be the knowledge building that Leitão (2000) and Andriessen et al. (2003a, b) name as an important goal of argumentation.

Concepts	Sequences	a particular activity or task is expressed as a series of individual steps or instructions that can be executed by the computer
Loops	a mechanism for running the same sequence multiple times	
Events	one thing causing another thing to happen	
Parallelisms	sequences of instructions happening at the same time	
Conditionals	the ability to make decisions based on certain conditions, which supports the expression of multiple outcomes	
Operators	provide support for mathematical, logical, and string expressions, enabling the programmer to perform numeric and string manipulations	
Data	involves storing, retrieving, and updating values	
Practice	Being Incremental and Iterative	an adaptive design process, one in which the plan might change in response to approaching a solution in small steps
Testing and Debugging	it is critical for designers to develop strategies for dealing with – and anticipating – problems	
Reusing and Remixing	Building on other people's work has been a longstanding practice in programming, and has only been amplified by network technologies that provide access to a wide range of other people's work to reuse and remix	
Abstracting and Modularizing	Abstracting and modularizing, which we characterize as building something large by putting together collections of smaller parts, is an important practice for all design and problem solving.	
Perspective	Expressing	computational thinker sees computation as more than something to consume; computation is something they can use for design and self-expression
Connecting	wide variety of ways in which an individual Scratcher's creative practice benefitted from access to others, through face-to-face interactions or online	
Questioning	computational perspective of questioning, we look for indicators that young people do not feel this disconnect between the technologies that surround them and their abilities to negotiate the realities of the technological world	

Table 1. Brennan and Resnick's (2012) computational thinking framework.

Methods

This research is rooted in the framework of the sociocultural perspective (Vygotsky, 1978; Wertsch, 1991). This study is part of a three-year ethnography (Hammersely & Atkinson, 2007) of both the online Scratch community and a three-year ethnographic study of Scratch workshops conducted in libraries in underserved neighborhoods in Los Angeles county. This paper focuses on the data collected from the online Scratch community. Observations were undertaken by two researchers. The researchers read both the data on the forums and the data in the studios and projects. The data collection of observational fieldnotes (Emerson, Fretz, & Shaw, 2011) for the online community took place from October 2014 to

October 2015, with roughly 500 hours of observation of the community. Data snippets of conversations were collected that illustrate learning and development within the community, this was along with descriptive fieldnotes. The long length of data collection was intended to get a large sample due to the immense size of the Scratch community and the amount of data available. The data was coded using qualitative coding (Saldana, 2012), using an a priori coding scheme made up of computational thinking perspective – expressing, connecting, and questioning (Brennan & Resnick, 2012) and Alagoz's (2013) argumentation framework. This analysis does not look at computational thinking of each individual participant but looks at the community at a collective intelligence (Levy,1999) level, taking a holistic view of the community. Collective intelligence is used as a lens for conceptualizing the community interactions as a whole.

Discussion

In Scratch, argumentation is used in a variety of situations, including in the forums, projects comments, and studios, with many Scratchers participating in all three settings. The forums are on average longer-form communication than the studio comments, which creates an atmosphere more conducive for quality argumentation. This analysis focuses on a lengthy and heated discussion that took place in the community in 2014, which started with a new thread that detailed a change to the community posted by a Scratch team member. This thread was selected because it was a polarizing event in the community, which made it very active and passionate. The first post in this thread announces that the numbers of Followers (those who a Scratcher follows) and of Following (those who follow a Scratcher) would no longer show on the profile page of Scratchers, with the exact number of followers still available by clicking "View All". The Scratch team member adds to the description "We want to make sure people see someone's projects, studios, and connections and not focus too much on the numbers. After all, it doesn't matter if you have 10 followers or 10,000. Scratch is a place for creating, collaborating, and supporting fellow Scratchers." This is the argument in support of the change made by the Scratch team, which set up the following argumentation from the Scratch community. The thread created a total of 769 posts before it was closed. The Scratch community was divided as to whether or not they supported the change. The reasoning for the divide and for their use of argumentation was influenced by their development of computational thinking. These are samples of the most common arguments for and against the change.

> Scratcher1: Ah, hopefully it will discourage follow4follow accounts. ? [Other]
>
> Scratcher2: That may be the whole point of removing the following/follower count.

I also want to add that the comment count maybe have been removed to get discourage messages like these: "800th Comment!" [Other]

These first two examples illustrate Other (which encompasses agreement, disagreement, submission, restatement, etc.), but they have an undertone of Expressing from computational thinking. This is because those who are interested in stopping follow4follow accounts (accounts that are willing to follow others in order to be followed) and number driven commenting, are part of a larger overarching argument in the Scratch community that these types of activities actually limit self-expression and creativity, because follow4follow accounts and number driven commenting decrease the quality of projects and accounts.

Scratcher3: Why I believe that this update is bad:

Taking the follow numbers away from the profile page does nothing. …This means that I'll hardly know when somebody unfollows me. I like to know if that happens because I don't want to be advertising something to repeatedly and annoying people, being too active… etc. However, about the people who are obsessed with followers, how will that discourage them? If they have enough time to spam several hundred people with follow requests, how is clicking one little button going to get them to stop? And taking the numbers away doesn't mean that they don't want followers any more. [Counter-alternative, Questioning, Connecting]

I like to keep track of everything on my profile and projects so that I know what people like, what types of projects encourage what types of activity, and so on. For example, I can tell you that my Updates project got quite a number of random comments on my profile, love its on my other projects, and not many love its for itself (compared to my other things of course). … If there were no numbers, then how would I know not to do that again? [Counter-alternative, Questioning, Connecting]

When you visit a profile, you usually have no idea how good or bad a user is at Scratch. Should you check out their projects? Should you not? Is the user just getting all of their thumbnails from another user that makes them look better at Scratch? Well, to determine this, many Scratchers look at the number of followers. If there are a lot of followers in a short amount of time, then you know that it's probably worth checking out their projects! I usually will compare the number of followers to the joined date so that I know if the given Scratcher is something new in the world of Scratch. I'm not saying that it's not worth checking out projects if the user doesn't have many followers, but I subconsciously do this a lot and it's really annoying if I need to click twice more to do this! That's a big waste of time! (Thanks to other Scratchers for reminding me of this). [Counter-alternative, Questioning, Connecting]

Scratcher4: Sure follows for follows can be annoying but being able to see each others followers also can be used as encouragement to try harder to make projects. Seeing someone with more followers than you have can encourage you to make more effort into your projects or at least get close to reaching the amount of follows that that person has. / And also, I don't want to have to load another page just to check my followers or who I'm following. [Counter-Critique, Questioning, Connecting]

Scratcher5: Well see, sometimes I just want to find a user with really good high quality games when I'm bored, depressed, or needs some relaxing time. I usually check the follower count to know how good the user is. If the user has lots of followers, that usually indicates his games are good and I check out the projects. That might seem biased but it really helps to find good projects. I still check out lower quality projects and low-follow scratchers to encourage them. I'm not trying to discourage others, I'm just saying that some scratchers make better projects than others and follower count helps me to find them. [Counter-Critique, Questioning, Connecting]

As can be seen from these data examples, those that offer critique present quality arguments, and it is these arguments that demonstrate computational thinking. These Scratchers are countering the original argument that was posited the Scratch team, using counter-alternative and counter-critique. In the same argumentation, as demonstrated by the sample, they support their argument with computational thinking using connecting and questioning, which is developed throughout use of Scratch (Brennan & Resnick, 2012). Connecting is the plethora of ways in which "an individual Scratcher's creative practice benefitted from access to others, through face-to-face interactions or online," (Brennan & Resnick, 2012). What is happening in this conversation is Scratchers are pushing back against a change to the platform because they perceive it as inhibiting their ability to find other Scratchers to connect to using what they see as a major marker of quality (i.e., the number of followers someone has). Questioning is also seen throughout this argumentation. Brennan and Resnick, explain questioning in computational thinking as looking "for indicators that young people do not feel this disconnect between

the technologies that surround them and their abilities to negotiate the realities of the technological world," (2012). In this argumentation the Scratchers demonstrate their connection between the technology that the platform offers, in this case the indication of number of followers, and their ability to negotiate the technological world. Scratchers are using both questioning and connecting as a way to support counter-critique and counter-alternative. They are specifically arguing that losing the ability to see the number of followers directly impacts their ability to judge if another Scratcher does quality work and if that Scratcher's advice is sound. They also posit that the change would affect their creativity, impacting their ability to find quality projects to be inspired by, thus connecting computational thinking with argumentation

Conclusion

Computational thinking being demonstrated within the Scratch community is not a surprise given that computation thinking was included in the design of Scratch, however the connection between argumentation and computation thinking was a surprise. As a result, I had to consider how computational thinking can improve argumentation. Using the Center for Computational Thinking's (n.d.) conception of computational thinking as a whole life not solely a computer science framework, this study sheds light on the Scratch community utilizing computational thinking that they develop in their use of Scratch to contextualize and support their argumentation. For Scratchers, the forum offers them a place to actively engage in a community they feel strongly about. The forum functions as a place where the Scratch community can express their opinions, argue for and against features of the community, and take an active role in the functioning of the community. These findings inform and expand the current understanding of computational thinking and how it is used by youth to support argumentation.

Acknowledgments

This research has been funded by the National Science Foundation Cyberlearning grant "Coding for All: Interest-Driven Trajectories to Computational Fluency".

References

Alagaoz, E. (2013). Social argumentation in online synchronous communities. *International Journal of Computer-Supported Collaborative Learning, 8*(4), 399-426.

Andriessen, J., Baker, M., & Suthers, D. (2003a). Argumentation, computer support, and the educational context of confronting cognitions. In J. Andriessen, M. J. Baker, & D. Suthers (Eds.), *Arguing to learn: Confronting cognitions in computer-supported collaborative learning environments*(Vol. 1, pp. 1–25). Dordrecht: Kluwer Academic Publishers. http://ses-perso.telecom-paristech.fr/baker/publications/ArticlesBakerPDF/2003/2003EtAl-a.pdf

Andriessen, J., Erkens, G., Van De Laak, C., Peters, N., & Coirier, P. (2003b). Argumentation as negotiation in electronic collaborative writing. In J. Andriessen, M. Baker, & D. Suthers (Eds.), *Arguing to learn: Confronting cognitions in computer-supported collaborative learning environments* (Vol. 1, pp. 1–25). Dordrecht: Kluwer Academic Publishers.

Bereiter, C. (2002). Education and mind in the knowledge age. Mahwah, NJ: Lawrence Erlbaum Associates.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. Proceedings of t*he American Educational Research Association (AERA) annual conference*. http://scratched.gse.harvard.edu/ct/files/AERA2012.pdf

Burke, Q., & Kafai, Y.B. (2012). The writers' workshop for youth programmers: Digital storytelling with Scratch in middle school classrooms. *Proceedings of the 43rd ACM technical symposium on Computer Science Education*, ACM 2012. https://www.researchgate.net/profile/Quinn_Burke/publication/230874961_The_Writers%27_Workshop_for_Youth_Programmers_Digital_Storytelling_with_Scratch_in_Middle_School_Classrooms/links/0912f5059e7c6a43ed000000.pdf

Center for Computational Thinking. (n.d.) *What is computational thinking?* Retrieved July 7, 2016 https://www.cs.cmu.edu/~CompThink/

De Moor, A., & Efimova, L. (2004). *An argumentation analysis of weblog conversations*. Paper presented at the 9th International Working Conference on the Language-Action Perspective on Communication Modelling (LAP 2004), New Brunswick.

Emerson, R. M., Fretz, R. I., & Shaw, L. L. (2011). *Writing ethnographic fieldnotes*. University of Chicago Press.

Erduran, S., Simon, S., & Osborne, J. (2004). TAPping into argumentation: Developments in the application of Toulmin's argument pattern for studying science discourse. *Science Education*, 88, 915–933.

Fields, D. A., Giang, M. & Kafai, Y. B. (2013). Understanding collaborative practices in the Scratch online community: Patterns of participation among youth designers. In N. Rummel, M. Kapur, M. Nathan, & S. Puntambekar (Eds), *CSCL 2013 Conference Proceedings, Volume 1*. International Society of the Learning Sciences: Madison, WI, 200-207. https://cdn.scratch.mit.edu/scratchr2/static/__50506d8f83d87df4175ef23ca9ada178__//pdfs/research/Fields_etal_CSCL_2013.pdf

Hammersley, M., & Atkinson, P. (2007). *Ethnography: Principles in practice*. Routledge.

Koh, K. (2013). Adolescents' information-creating behavior embedded in digital media practice using Scratch. *Journal of the American Society for Information Science and Technology. 64*(9), 1826-1841. http://onlinelibrary.wiley.com/doi/10.1002/asi.22878/abstract

Kuhn, D. (1991). *The skills of argument*. Cambridge: Cambridge University Press.

Kuhn, D. (2005). *Education for thinking*. Harvard University Press: Cambridge, MA.

Leitão, S. (2000). The potential of argument in knowledge building. *Human Development, 43*(6), 332–360. http://www.karger.com/Article/Abstract/22695

Lévy, P. (1997), Collective Intelligence: Mankind's Emerging World in Cyberspace, Perseus, Cambridge, MA.

O'Connor, M.C. & Michaels, S. (1996). Shifting participant frameworks: Orchestrating thinking practices in group discussion. In D. Hicks (Ed.), *Discourse, learning and schooling* (pp. 63-103). Cambridge: Cambridge University Press.

Rourke, L. & Kanuka, H. (2007) Barriers to online critical discourse. *International Journal of Computer- Supported Collaborative Learning, 2*(1).

Saldaña, J. (2012). *The coding manual for qualitative researchers.* Thousand Oaks, CA: Sage.

Schiffrin, D. (1987). *Discourse markers*. Cambridge: Cambridge University Press.

Scratch. (n.d.) *Statistics.* Accessed on 3/14/2016, https://scratch.mit.edu/statistics/.

Steinkuehler, C. (2007). Massively multiplayer online gaming as a constellation of literacy practices. *eLearning, 4*(3) 297-318.

Toulmin, S. E. (1958). *The uses of argument*. Cambridge: Cambridge University Press.

Vygotsky, L. S. (1978). *Mind and society: The development of higher mental processes*. Cambridge: Harvard University Press.

Walton, D. N. (1996). *Argumentation schemes for presumptive reasoning*. Mahwah:Lawrence Erlbaum.

Wertsch, J. (1991). *Voices of the mind: A sociocultural approach to mediated action*. Cambridge: Harvard University Press.