

Simulating Facebook's Newsfeed for Writing Pedagogy

Daniel Libertz, University of Pittsburgh

In this paper, I describe a social media newsfeed simulation I made to get students thinking and talking about algorithms from three rhetorical perspectives: as persuasive (Beck, 2016), as enacting constitutive rhetoric that construct subjectivities (Charland, 1987), and as audience (Gallagher, 2017). These distinct but related rhetorical characteristics of algorithms are complex and can be difficult to think about in isolation and in relation to one another through traditional rhetorical and English pedagogies (e.g., reading and discussion, writing in response to a prompt), but I argue here that the simplicity and dynamism of a simulation can more effectively help students see themselves as rhetoricians that work with, against, and as composers of algorithms.

As an introduction to rhetorical and procedural characteristics of algorithms and social media newsfeeds specifically, I developed a simulation that helped students see themselves in several roles: as a subject constituted (Charland, 1987) by the algorithm, as a writer *to* an algorithm in terms of algorithmic audience (Gallagher, 2017), and as producer of an algorithm and thus how they might persuade algorithmically (Beck, 2016). In the following section, I explain the pedagogical benefits of simulations that model a procedural rhetoric. I then go into more detail on the three rhetorical characteristics that the simulation attempts to enact. Finally, I walk through how the simulation was designed and how I taught with it.

Procedural Rhetoric: Game and Simulation Pedagogies

Ian Bogost (2007) explained that procedural rhetoric is an argument or expression made by processes. In other words, the way in which a process takes an entity through a series of operations or actions governed by rules set up to ensure what is and is not possible creates affordances and constraints that are persuasive insofar as they facilitate what a user experiences, and, thus, can influence what a user thinks or believes. Games, as reliant on processes, are useful pedagogical tools grounded in procedural rhetoric that can help learners to understand processes in the real world. Games have also long figured as useful frameworks for teaching rhetoric and composition. Rebekah

Shultz Colby (2017) referred to games as useful pedagogical moves because of their multimodal affordances and constraints, their rule-based systems that can promote exploration of modelling capabilities, and the genre ecologies they participate in (e.g., design texts, paratexts). Shultz Colby also notes that games help teach systemic thinking, often in an enjoyable fashion. Simulations, too, can achieve the same goal, albeit in slightly different ways. To understand how, it is important to think about the differences between games and simulations.

Louise Sauv   et al. (2007), based on a literature review of nearly 2,000 articles about games and simulations, concluded that the main differences between games and simulations are that games are nearly always fictitious, rule-based environments that center around a conflict and/or goal to achieve whereas simulations attempt to represent reality in a simplified and dynamic fashion, modeling reality without a goal or conflict driving the action. Simulations are often used in educational settings emphasizing technical expertise in professions like nursing, piloting, the military, etc. Other instances might be a case study approach to highlight ethical complications—in writing pedagogy, one example of this is Christy Friend and Mark Minkster’s (2002) simulation assignment on ethical argument that focused on admissions decisions in the context of affirmative action. For the most part, though, game approaches appear more prevalent in writing pedagogy than simulations.

While I’m not completely convinced by the usefulness of the distinctions between games and simulations as hard and set (for instance: does an airplane simulator count as a game since it has a “goal” of reaching its destination safely?), I do appreciate the description of the simulation in how it underscores how lived experience is attempted to be modeled but simplified in service of dynamic experimentation by users. The experiential nature of a simulation can aid in bringing energy and accessibility to conceiving digital spaces driven by algorithmic processes, like that of the newsfeed, as a rhetorical environment. Previously, I had struggled to teach newsfeed algorithms as rhetorical with conventional means—namely, having students write blog posts in response to a reading on algorithms (i.e., Zeynep Tufekci’s wonderful 2015 piece on “algorithmic gatekeeping”) —and I believe that is because it is difficult to rely solely on traditional literate practices when engaging rhetorics reliant on procedure.

While algorithms are processes in the real world worth exploring, I wanted to have students explore these processes *as rhetoric*. Algorithms are persuasive procedurally (Brown, 2015; Holmes, 2014)— the processes they set in motion help to facilitate what a user experiences, and this facilitation can impact what a user thinks or believes. I wanted students to study this type of persuasion by critically and realistically experiencing it. Instead of using a

game as a procedural rhetoric to explore something about a concept or idea (e.g., see Bogost's explication of *The McDonald's Videogame* as an argument about capitalism), I made a procedural rhetoric (a simulation) to learn about a procedural rhetoric (newsfeeds). Rather than abstract away by creating a world via a game, I wanted to concretize toward a more accessible version of a realistic newsfeed environment via a simulation. The simulation's advantage is that it can utilize the rhetorical nature of processes pedagogically by creating a transparent, simplified, and dynamic experience for students while also providing a common experiential object that could anchor an exploration of algorithmic rhetoric for the class.

In the next section, I outline in more detail the three rhetorical characteristics of social media newsfeed algorithms considered through the simulation: their nature as "quasi-agents" that persuade, their execution of constitutive rhetoric, and their role as an audience.

Three Rhetorical Characteristics Explored by Simulation

Estee Beck (2016) argued that algorithms are quasi-agents that "carr[y] forward the agency of human symbolic action" through how they organize the world for human and computer interaction, how they systematically include and exclude information, and how they are ideological insofar as they are inscribed with knowledges and biases of their creators. Most obviously, a social media algorithm includes and excludes certain information to drive human decision-making along a certain path (i.e., staying on the website to be fed more ads). Since algorithms, from this standpoint, are persuasive in themselves, a simulation can help model how social media newsfeeds *might* work since most are blackboxed, proprietary entities. A simulation helps model the dynamism of an aspect of reality while simplifying it enough for students to walk through and experience to assist learning. As opposed to an activity like having students track patterns in their own social media accounts, a simulation can become a shared text that a classroom can try to figure out together, while *also* finding the answer to help validate their suspicions about how newsfeed algorithms are rhetorical.

Another rhetorical characteristic of newsfeed algorithms that I wanted to have students think about was constitutive rhetoric. Maurice Charland claimed that constitutive rhetoric works by interpellating subjects that have always already been constituted as subjects through a "series of narrative *ideological effects*" (p. 134). Three important elements are at play here: establishing a collective identification, the "positing of a transhistorical subject" (i.e., beyond the living to include the dead; timeless subject), and the illusion of freedom (i.e., since the ideological narrative is written, once one takes part as a

constituted subject, the outcome can only be what the narrative has set forth).

Constitutive rhetoric in a digital context looks different than the public address context that Charland was working from. Shira Chess (2018) argued that the construction of “casual” gamers and “hardcore” gamers were evidence of constitutive rhetoric in digital contexts. Design and marketing choices helped to constitute the “idealized woman gamer” for “casual” games like *Kim Kardashian: Hollywood* and *Restaurant Story*. These games used narrative and subtext to hail a subject that is white, heterosexual, cis-gendered, middle-class, able-bodied, slender, and typically a mother. Chess maintained that her own history as a gamer has been conditioned by the material conditions from which casual games are made and have helped constitute her as the idealized market for casual games. The technologies and the market conditions surrounding them interact with human actors to produce these sorts of rhetorics. In the context of social media newsfeeds, this happens even more implicitly than in Chess’s case.

The technological complications of algorithms as “quasi-agents” instills a constitutive rhetoric by procedure rather than (only) linguistically. A subject is hailed by the posts and ads generated as users scroll through their feeds, and their interactions with these objects (among other things, like the tracking data that social media platforms purchase from web browsers) creates a subject. The rhetorical act of including and excluding certain posts forms an online subject that would fit a bucket for advertisers. For Facebook, we can see this quite clearly in “Your Categories” under “Your Information” on the “Your ad preferences” page, where such categories like “liberal” or “multicultural affinity” are populated in the service of ad targeting. As users see and interact with more posts and ads, users fulfill the “illusion of freedom” ideological step in constitutive rhetoric by continuing to engage in predictable ways, or, as Kevin Brock and Dawn Shepherd (2016) might put it, they complete the enthymeme. There are nefarious ends to this, of course: fulfilling and re-affirming identity markers in this way helps set up possibilities of exclusion and inclusion that are discriminatory. The National Fair Housing Alliance (2018) offered evidence of just this in their findings about the possibilities for excluding certain groups of people seeing ads for housing. Additionally, the realities of the “filter bubble” (Pariser, 2011) can be exacerbated when identities are constituted and re-constituted in these ways.

Finally, in addition to algorithmic persuasion and the constitutive rhetoric enabled by advertising realities of social media, a final important rhetorical characteristic of social media algorithms I wanted to focus on with students is conceiving algorithms as an audience. John R. Gallagher (2017) made a compelling case that students need to be aware of how to write to an “algorithmic audience”—that is, writers should be aware of how to write to a set

of procedures that prioritize some content over other content. For instance, Google's *Search Engine Optimization (SEO) Starter Guide* informs writers that anything from an appropriate length for a page's description meta tag to tips on how to write hyperlink text impacts possibilities for search hits. For a social media algorithm, how often something is liked or favorited or the utility of commenting on an older post with many likes has direct consequences for a post's circulation.

According to Gallagher, "[t]eaching this type of awareness and habit means that in addition to teaching students to write for a particular set of readers, we are also teaching for a set of procedures that highlight content for readers" (pp. 26-27). This is an important distinction for digital writing: it is not only important to attend to how students write with digital vs. print media (e.g., attending to a comment function for a blog post, using hyperlinks, ability to incorporate multiple modes more seamlessly), but we must also prioritize teaching circulation in digital contexts through the notion that algorithms are an important audience in need of suasion to ensure that a rhetor's content is circulated in ways beneficial to their rhetorical goals.

By working through the simulation, students can see algorithms as rhetorical from multiple perspectives and, in the best case, become more critical readers of algorithmic rhetoric, thoughtful writers composing *with* algorithms, and also begin to see the possibilities afforded in *being* algorithmic rhetors. In this way, students can accomplish something akin to James J. Brown, Jr.'s (2015) notions of arguing *with* software and arguing *in* software; software is a tool we necessarily must use rhetorically (with social media as no exception here), an interlocutor that we must contend with to reach our rhetorical goals, and it provides an environment of multiple audiences we cannot escape (our data are always welcomed, tracked, and engaged with in ways that shape what a newsfeed can be and how we interact with it). As an "ethical program," a newsfeed algorithm governs what is and is not possible in the networked infrastructure of the internet, structuring user relations by responding over time to interactions between users, software, and other users.

Designing and Teaching the Newsfeed Simulation

By providing a learning environment where students could systematically test hypotheses by responding to posts in different ways (e.g., liking a post, ignoring it, commenting, sharing) for a simplified system interpreting those interactions, my students and I were able to leverage the procedural rhetoric of the simulation to explore all three rhetorical characteristics of algorithms described above. In the simulation, I wanted students to view posts, interact with those posts, and have the simulation use those inputs to produce diverse

outputs. I also wanted students to actually see the code to help display a more transparent experience than blackboxed experiences typical of most interfaces. I could have used, say, Google Forms or SurveyMonkey to simulate a newsfeed by using question logic, but I thought allowing students to see code helped to demystify the notion that algorithmic rhetoric is inaccessible—I certainly can't replicate anything very sophisticated as a (very green) novice, but the essential idea of this practice as rhetorical is interdependent with how code works. This way, the material reality of code is front and center to keep in mind how interfaces hide their underlying procedural rhetoric that influences reading and writing in social media contexts.

I wrote the simulation with Python, using [Jupyter Notebook](#). Jupyter Notebook allows users to edit and run documents from a web browser, which can be useful for student interaction. To ensure that students could easily access and run the program, I also used [Binder](#), an application that allows any user to open a Jupyter notebook in an executable environment. If you want to try for yourself, go to [my GitHub repository](#) and click the “launch binder” button toward the bottom of the screen— this will link you to your own version of a Jupyter notebook of the program. I'll explain how the program works below, but I encourage you also to run it for yourself to supplement the following explanation.

I decided to use Facebook as the social media platform model for the simulation because of familiarity and the timeliness considering recent news about Cambridge Analytica. I then decided to focus on clicks, likes, comments, and shares as the inputs. While the system I built is rather simple and is not at all how Facebook works—neither according to the old EdgeRank algorithm nor with Facebook's new system dependent on machine learning (which, of course, is blackboxed)—it does provide a simple way to introduce students to one possibility among several of how an algorithm could filter posts in a newsfeed—rhetoric, after all, is about choices and possibilities. I have taught this lesson using the simulation three times, making slight tweaks each time, which I will note in the below, when relevant, as I describe the stages of the lesson and account for some student reactions to it.

The first step in the simulation is to decide whether to click on an article in a post, click and like it, or take no action and keep scrolling; the user is then given an option to comment on the article or to provide no comment. A score is generated from the user's decisions and stored for this first post in the simulated newsfeed, and the output is displayed after the user engages. The user then has two more posts in their newsfeed and is prompted to make the same decisions.

Based on user engagement with three posts in their newsfeed, the user is given a cumulative score which then delivers a fourth post in the newsfeed

based on that score. One of three possibilities would occur: a post sharing an article about transgender people in the military, a post sharing millennials' negative impact on the economy, and a post sharing a clickbait article on how to get a job. The user is then prompted to engage with this new individually-tailored post in the same way as before, and this score is added to the cumulative score. Based on this score, the user is given a batch of four new posts in their newsfeed and they are prompted to choose one to share and add some of their own text to accompany the shared post.

Since all students are working from the same simulation while also engaging with it in different ways, they can now mine their experience with the simulation to discuss why their results were different. I ask students to pair up to compare their results and to theorize how the newsfeed functions, followed by a large group discussion about these theories. To avoid a “gotcha” situation (i.e., students feeling pressured to find the “right” answer for how the program works), I preface the discussion by stating that since an algorithm at its most basic level is a set of procedures reliant on inputs to create outputs in service of a task or solution to a problem, almost any theory about why you saw certain posts could have an algorithm built to support it.

Some answers I have received at this stage of the lesson in the past organically address two of the three rhetorical characteristics I outlined above: algorithms as persuasive and algorithms as producing constitutive rhetoric. For instance, one student pairing theorized that since they both had high engagement with a post that shared a video on how to make mac and cheese, that they then got posts with more progressive politics because “millennials don't cook much and would be receptive to easy recipes” and millennials have more progressive politics. Here, the reasoning was that the program was interpreting a certain culinary interest that correlates with political positions—something that very much could exist, especially in the context of machine learning. Through processes of receiving inputs and generating new outputs, there is a claim made about the world in such a way made by inclusionary and exclusionary moves made in this environment, thus depicting the newsfeed algorithm as persuasive. I also have tried to push students to think about this as a constitutive rhetoric. Going back to the mac and cheese example, we explored in class discussion how, over time, as the posts assemble together, the collection of these posts hail the account user to be a subject who enjoys trendy-but-simple recipes aligning with certain political positions to form something called “millennial.” This transhistorical subject completes a narrative about seemingly mundane cultural tastes and liberal or leftist political views: of course you'd like this food, people who like this food hold these beliefs, as millennials. Thus, during discussion, students were able to contribute to an understanding of newsfeeds as rhetorical from both of these perspectives.

After some discussion aimed at linking the idea that algorithms persuade by inclusion and exclusion and how they play a role in asking users to fulfill a subject position by a cascading assemblage of generated posts, I walk through some background code that was imported into the simulation to show how it worked (see the folder “Gravy” in the Jupyter notebook). Essentially, all posts in the simulation were flagged as politically left, politically right, or “neutral,” and the level of engagement would contribute to a score that would correlate with one of those three buckets. Students get to see how it works, but the point underscored is that this is one of an infinite number of ways we could have created procedures to filter posts—in other words, the choices in design are rhetorical choices as much as they are computational, mathematical, logical, etc.

The discussion about how I made this simulation also helps to display Beck’s (2016) point about how a creator’s ideology seeps into the algorithm. For instance, I ask students to do a think-pair-share on a few questions: what counts as a “neutral” post? And how does one fairly decide how to score those posts to best align with that classification (I decided to use a random number generator for neutral posts)? How can you decide what is counted as left or right? Did bias creep into this simulation (e.g., a post about millennials negatively affecting the economy was scored as a “politically right” post, but surely generational critiques are not limited to conservatives)? One of the goals here is to get them to think about how decisions to put what object in which bucket is a rhetorical problem. And further, that the step to even create these buckets implies that the world is accurately organized by such terms, which, of course, is problematic.

The next portion of the lesson asks students to run the simulation a second time to consider how to write *for* the algorithm. Using what we know about how the simulation works, students are asked to think strategically about the last step in the simulation of choosing an article to share and how to caption that share (and thus write for the newsfeed algorithm). What sort of language would be flagged for the right bucket that you want your post to circulate toward? How do you entice the click? At this stage, doing the writing in class helps us discuss important aspects of social media writing. Discussions have engaged topics of how to say just enough but not too much, avoiding a TL;DR situation, considerations of the ethics of overpromising (i.e., clickbait), and to avoid possible clichés or to embrace them depending on your audience (e.g., using all caps). In some senses, these are considerations of human audiences, but by placing this discussion within an activity about writing text within the environment of the simulation, students can more clearly see how this is also a matter of algorithmic audiences when considering the potential for filtering some posts over others based on likes, comments, and other metrics.

The very last step in this lesson is to have students think about revising the simulation. The third time I've taught this lesson, I asked students literally to do this by revising the Python code, but it is just too much for one lesson—both in terms of time but also considering complications of the restrictions set in for students who are both unfamiliar with programming or students that are advanced programmers (in many cases *very* advanced!). What works better is what I did the first two times teaching with the simulation: having students create a mock-up or use pseudocode to create a different system of how posts are divided up or how they are scored differently to open up more possibility for big picture thinking within the time limits of the lesson.

I start them with an example, to model one possibility: instead of trying to score by one of three political buckets, another way to organize what posts would be filtered is by dividing up the alphabet into thirds and aligning posts by their first letter with each bucket. I pass out a handout that lists the posts in the simulation by title, prompting students to become algorithmic rhetoricians. This stage asks students to consider how algorithms are systematic organizers of the world that work based on inclusion and exclusion with inevitable ideological infiltration by human agents into predictable yet at times surprising quasi-agents that replicate certain ideologies thus inscribed. For example, students in the class have divided the posts by length of post title to filter them based on engagement for three new buckets. Another group of students tried to score “hate-clicks” as a way to filter certain posts (e.g., if a certain user has thus far been politically left, their click of a politically right post would be given a score that further puts them to the left rather than to the right, and gives them a certain kind of inflammatory political right post in the future). By providing students space to think about other possibilities for the newsfeed's code, students get to think carefully about multiple means of persuasion for an algorithmic rhetorician, and therefore, think more critically about the space of the newsfeed as a multifaceted rhetorical space/rhetorical agent.

Simulations provide pedagogical benefits by using a simple model of a more complicated reality of procedural rhetoric—with dynamic possibilities in outcomes when playing through it—to introduce students to thinking about how algorithms are rhetorical objects worthy of our attention and respect. I hope that this sort of pedagogy represents a small contribution to a possibility of what Annette Vee (2017) called coding literacy, in that students see the code, try to understand it, see what it can do, and reflect on how things can be done differently—thus, through such material and social engagement, claiming some form of power back from the elite class that monetizes software like newsfeed algorithms. Simulations can be used as introductory materials to how programming impacts our literate practices in all sorts of ways, how we are all writing “with” and “in” software constantly.

Acknowledgements

I want to thank Matthew Lavin for his help planning this lesson in the early stages and his feedback on the functionality of the program. I also want to thank Annette Vee, whose suggestions after observing an iteration of this lesson were invaluable. Finally, thank you also to the reviewers for their helpful and kind feedback.

References

- Beck, Estee. (2016). A theory of persuasive computer algorithms for rhetorical code studies. *Enculturation*, 23. Retrieved from <http://enculturation.net/a-theory-of-persuasive-computer-algorithms>
- Bogost, Ian. (2007). *Persuasive games: The expressive power of video games*. Cambridge, MA: MIT Press.
- Brock, Kevin & Shepherd, Dawn. (2016). Understanding how algorithms work persuasively through the procedural enthymeme. *Computers and Composition*, 42, 17–27.
- Brown, James J., Jr. (2015). *Ethical programs: Hospitality and the rhetorics of software*. Ann Arbor: University of Michigan Press.
- Charland, Maurice. (1987). Constitutive rhetoric: The case of the *Peuple Québécois*. *Quarterly Journal of Speech*, 73(2), 133–150.
- Chess, Shira. (2018). I am what I play and I play what I am: Constitutive rhetoric and the casual games market. In Aaron Hess & Amber Davisson (Eds.), *Theorizing digital rhetoric* (pp. 224–233). New York, NY: Routledge.
- Friend, Christy, & Minsker, Marc. (2002). Merit vs. diversity? A simulation exercise introducing students to ethical arguments. *Kairos: A Journal of Rhetoric, Technology, Pedagogy*, 7(2). Retrieved from <http://kairos.technorhetoric.net/7.2/binder.html?sectiontwo/friend/Index.html>
- Gallagher, John R. (2017). Writing for algorithmic audiences. *Computers and Composition*, 45, 25–35.
- Google. (n.d.). Search engine optimization (SEO) starter guide. Retrieved May 9, 2018 from <https://support.google.com/webmasters/answer/7451184?hl=en>
- Holmes, Steve. (2014). Rhetorical algorithms in bitcoin. *Enculturation*, 18, 1–16. Retrieved from <http://www.enculturation.net/rhetoricalalgorithms>
- National Fair Housing Alliance. (2018, March 27). Facebook sued by civil rights groups for discrimination in online housing advertisements. Retrieved May 9, from <http://nationalfairhousing.org/2018/03/27/facebook-sued-by-civil-rights-groups-for-discrimination-in-online-housing-advertisements/>
- Pariser, Eli. (2011). *The filter bubble: What the internet is hiding from you*. New York, NY: Penguin Press.
- Sauvé, Louise, Renaud, Lise, Kaufman, David, & Marquis, Jean-Simon. (2007). Distinguishing between games and simulations: A systematic review. *Educational Technology and Society*, 10(3), 247–256.

- Shultz Colby, Rebekah. (2017). Game-based pedagogy in the writing classroom. *Computers and Composition*, 43, 55–72.
- Squire, Kurt. (2011). *Video games and learning: Teaching and participatory culture in the digital age*. New York, NY: Teachers College Press.
- Tufekci, Zeynep. (2015). Algorithmic harms beyond Facebook and Google: Emergent challenges of computational agency. *Colorado Technology Law Journal*, 13(2), 203–217.
- Vee, Annette. (2017). *Coding literacy: How computer programming is changing writing*. Cambridge: MIT Press.