

Kyle Booten

Flusser's Demon:

Writing Under the Eye of an Automatic Critic

Machine Writing vs. Machine Editing

Writers tend to get the lion's share of fame; their editors, except in rare cases, languish in relative obscurity. This inequality of prestige is as true in the case of non-human writers and editors as it is in the case of human ones.

Since Christopher Strachey's (1954) computer-generated love letters and Theo Lutz's "Stochastische Texte" (1959) poet-programmers working in the mode of what is often called "electronic literature" or "littérature numérique" have produced countless verse-limning algorithms relying on a wide range of techniques, from the simple (e.g. mere randomization of words and phrases) to the complicated (e.g. artificial neural networks). No doubt the poet-programmers who practice this art will continue to discover innovative ways to instruct their machines to produce stories, poems, and other texts. At this point, however, the activity itself is familiar and well-established, with prizes and presses dedicated to it and with cultural institutions beyond the tight-knit "e-lit" world recognizing it. (Annie Dorsen's 2019 MacArthur "Genius" Grant, bestowed in recognition of her algorithmically-generated theater, is perhaps the current high-water mark.)

Anyone who has ever encountered an automatic spell checker knows that algorithms, in addition to themselves writing, can also *edit* the writing of humans. In general, poet-programmers have been somewhat more hesitant to design systems that interfere with human writing. When they do, these systems are typically presented as "co-authors." For instance, Robin Sloan's *rnn-writer* (2016) uses a neural network to suggest words and phrases in the flow of fiction writing. Poet-programmers tend to eschew systems that, like any spell checker, are explicitly pedagogical or that imagine the human writer's text to possess some deficit that could be rectified by algorithmic intervention.

Why is composing an algorithm to generate text a common artistic practice and composing an algorithm to correct human text a rare one? Perhaps this reflects a desire to avoid the annoyance that can be caused by misguided algorithmic advice, such as the kind proffered by "Clippit" (commonly known as "Clippy"), Microsoft's famously meddlesome word processing assistant. But the core difference between these two types of algorithms—the writer and the critic—is that they imply

different relations of power between the human and the program. The well-known practitioner of text-generation Ross Goodwin has called himself a “writer of writers” (Goodwin 2018), an appellation that points to the human programmer’s magisterial position above the algorithmic writer and, in turn, whatever is written. By remaining in charge of the algorithm (writing it, unleashing it, selecting from its outputs), the poet-programmer takes up a position analogous to the one that they would enjoy should they end up working as a software engineer for Google, Facebook, Uber, or any such company. In the vocabulary of McKenzie Wark (2015), this is the position of the “hacker” allied with the “vectoralist,” the capitalist who uses the infrastructures of digital media to control world at a safe distance from it. The notion of an algorithmic *critic* instantiates the opposite relationship between human and machine; here the machine has the power. The human becomes more like the Uber drivers who toil “below the API” (Rao 2017), bossed around all day not even by human overseers but by pushy, often stilted algorithms.¹

Given the enticing position of power held by the poet-programmers who “write writers,” perhaps it is not surprising then they have largely eschewed programming algorithms that will critique or edit human writing. However, this paper makes the case that that poet-programmers should think less about programming *writers* and more about programming *editors* and *critics* that will place in judgement a human-writer’s composition—this notwithstanding the fact that doing so, slipping “below the API,” may represent a threat to the writerly ego.

I make this case by way of Vilém Flusser’s theorization of the relationship between human and algorithmic creativity, specifically his unwillingness to entertain much daylight between the two. In *Into the Universe of Technical Images* (2011b), amidst a larger speculation about the nature of creativity in an age of artificial intelligence far more advanced than our own, Flusser pauses to muse on the potentials for “artificial critics” to reshape culture. The algorithms he imagines are no mere spell checkers; rather Flusser seems confident that algorithms will one day police not just “so-called value-free information” (such as whether a word is spelled correctly, or whether an Uber driver has arrived at a destination) but also matters of “ethical, political, and aesthetic information” (2011b: 177). Flusser quickly accelerates to a vision of a future in which certain algorithms create and other algorithms critique their creations, forming a closed loop of cultural production that sidelines the human. Still, his thoughts on the capacity of non-human entities invite us to imagine a stage that is perhaps not so far away, a future in which the human writer has not been totally excluded but is subjected at

¹ I further discuss this power relation elsewhere (Booten 2019a).

every turn to computational oversight—in which getting published in any respectable literary journal will require pleasing not a human editor with taste but rather a statistical model of “good poetry.”

But what exactly are these “automatic critics and censors” that Flusser imagines (2011b: 177)? How would they read poems? What would “beautiful” mean to them? By what standards would they issue their rejection letters? These questions are not trivial, since Flusser’s vision of it is as hazy as it is provocative, avowedly-utopian as it is technically precise. As a way of beginning to answer them, this paper also describes my attempt to design and create my very own automatic critic, a piece of software that would (in theory) help me to write better poems.

Maxwell’s Demon

Flusser’s *Into the Universe of Technical Images* ostensibly concerns visual rather than purely-linguistic media, yet his account of the “technical image” and the “telematic society” seems to apply just as well to what is read as what is seen. In fact, it is fairer to say that his theorization concerns neither image nor text but rather *information*.

For Flusser, the tenets of information theory as well as thermodynamics map directly onto human culture. According to his formulation, human culture is unlike nature in the sense that the latter tends toward “entropy” while the former does not. Entropy is a tricky term, especially since Flusser adapts it in a way that conflates its somewhat different meanings in information science and thermodynamics.² As is the case in thermodynamics, Flusser uses entropy to signify the dissolution of order. The second law of thermodynamics points towards the eventual point at which all energy will be evenly distributed—the total destruction of any order. Entropy’s inevitable but nonetheless catastrophic vanquishing of all order is acted out in miniature each time a cup of coffee cools to the temperature of the room, erasing the energetic distinction between the two. But Flusser also draws on Shannon’s information theoretic definition of entropy as a value that explains how predictable information is within a system. More entropy means less redundancy, more possibilities, more surprise. Another way of expressing the informatic definition of entropy is the average number of bits (a 0 and a 1) required to encode a message in a system. The entropy of a fair coin flip is one bit—a 0 for heads, and 1 for tails. But encoding a message describing the suit of a card drawn at random from a randomized deck of cards would require on average two bits: 00 for clubs, 01 for diamonds, 10 for

² For a thorough discussion of the history of the term entropy in both thermodynamics and information theory, see Hayles (1990).

hearts, and 11 for spades. The entropy the cards is higher, so you would have a better chance of guessing the result of a coin toss than the suit of a card chosen at random. (The entropy of the cards' face values is higher still.)

Blurring the thermodynamic and informatics senses of “entropy,” Flusser asserts that human culture resists the entropic drive through preserving memory via media of all kinds. Any mediated memory militates against the fact that our human brains all eventually die, losing whatever memories they held and yielding to the physical disorder of decay. Flusser does not seem to worry, as many have, that the increase of texts—be they digital emails or printed newspapers—represents an entropic deluge of information (see Hayles 1990). On the contrary, digital media have a special role to play in the war against universal entropy because they are not merely static but active and thus can *select* as well as *preserve* culture. In other words, algorithms can be “automatic critics” as well as “censors.”

Flusser illustrates the connection between entropy and criticism through a reinterpretation of the famous thought-experiment known as “Maxwell’s devil” (also known as “Maxwell’s demon”). The second law of thermodynamics leads us to predict that, when two vessels—one containing hot molecules, the other cold—are connected into a closed-system, the hot and cold molecules will even out between the two vessels. As the hot vessel gets colder and the cold vessel gets warmer, entropy obliterates the orderliness of the system. According to the thought-experiment developed by British scientist James Clerk Maxwell³, there could be a way around this inevitability. A “demon” graced with super-human perceptual powers could re-introduce order into the system. Standing guard at the portal between the two vessels that have reached the same temperature, the demon could selectively block the passage between the two so that all the hot molecules eventually would migrate to one side while all the cold molecules would be imprisoned in the other. Having cooled to the same temperature, the two vessels would, by the power of the demon’s magic, diverge in temperature. Entropy would be subdued, order created *ex nihilo*.

Maxwell’s demon is a kind of algorithm or “bot” that resists entropy within a physical, thermodynamic system. Flusser’s two-fold intuition is that 1) critics are also, in a similar way, demonic filters that resist entropy in cultural systems and that 2) this function could be automated by “automatic critics” that filter texts based on some linguistic criterion. It is worth pausing to add a bit more clarity to the analogy between critics and Maxwell’s demon that Flusser himself provides. Imagine a world in which all editorial decisions – should this article/novel/poem be published? – are made randomly by editors who simply draw lots rather than go through the trouble of discussing the rela-

³ Again I recommend Hayles’ (1990) consideration of entropy and of Maxwell’s demon in particular.

tive merits of submissions. This situation would be akin to the two vessels, connected by a tube, that have reached a thermal equilibrium, a state of disorder; just as a fast-moving particle may be just as likely to appear in the left vessel as the right, so in the case of truly random editorial decisions would a text that is (by some criterion) “good” be just as likely to go published or unpublished. What real, thoughtful editorial decisions do, at least in theory, is create some meaningful order between the “vessel” of non-published works and the “vessel” of published works. If you want to read “good” rather than “bad” literature (or history, or scientific treatises, etc.), you should probably pick from those texts that were selected for publication rather than those that were rejected, so long as you agree more or less with the criterion for inclusion. A human editor’s non-random sorting lowers the entropy of the closed system of all texts by introducing meaningful separation between them.

By what criteria, though, does Flusser imagine an automatic critic to “filter” the words and images we create? Here Flusser swings back from thermodynamics to information theory: he thinks that the role of “telematic” (read digital) media is to “distinguish between redundancy and information” and “decide in favor of information” (2011b: 116). In information theory, to say that a message is “high-information” is to say that it is unlikely.⁴ An algorithmic censor might readily deploy the criteria of information theory in order to sort out culture: “[A] German text is to be measured for its information content. In German, *X* is a rare letter, and the more frequently it occurs in the text to be measured, the more informative the text would become at the level of letters. The text could also be measured at the level of words, sentences, rhythms, or styles, however, without introducing any criterion other than rarity.” (Flusser 2011b: 118)

Here Flusser seems to imply an editorial protocol that would select “exciting” or “surprising” texts rather than linguistically unsurprising ones. This is not an unreasonable intuition. No writer (other than Kenneth Goldsmith or Pierre Menard) would expect to publish as their own certain famous poems or novels that already exist. In addition to being a possible violation of copyright, such a publication would be redundant. Likewise, a text need not be an exact replica of a previous text to deserve critical scorn. It is enough for it to be “cliché” or “derivative.” And perhaps it seems more literary to say that the sky is “hump-backed” rather than just “cloudy.” Insofar as human editors are always on the lookout for something new and vibrant, insofar as novelty is indeed a literary virtue, Flusser’s automatic editor would approximate it. Yet perhaps “approximate” is the wrong word, as it

⁴ For instance, if you were to be told that the words “the” and “and” appear in a text, you would have very little knowledge of—information about—what topic the text concerns. This is because these words are common to almost all texts in English. However, if you learned that the text contained the words “decant” and “Cabernet,” you would be able to make a much better guess regarding the text’s topic. These rarer words would carry more information.

implies that human cognition possesses some characteristic that, even if computers can mimic it, is not itself equivalent to algorithmic logic. Rather Flusser entertains the radical, disorienting *equivalence* between human and algorithmic “thinking.” In *Does Writing Have a Future?* (2011a), Flusser also resorts to the language of information theory, this time to offer a definition of poetry (which includes but is not limited to verbal art). Poetry offers us new “models of experience,” ways to see and experience the world (74). Poets may think that they produce these models through hard work or genius, but Flusser implies they flatter themselves. In reality poetry is “the computation of existing models with intrusive noise” (73). Take a normal way of using words, pass it through a “noisy channel” so that it is disrupted, and—voilà—poetry. This “noisy channel” may be a poet’s mind but, as Flusser’s deflationary definition implies, the noise might even be added through random algorithmic changes. A poet is just a machine for producing linguistic entropy. (We might add that, as entropy-machines, poets are inefficient, since they often require not just the basic equipage of food, housing, etc., but also extensive time for contemplative musing as well as expensive education.)

However, entropy is a subtle concept, and Flusser’s account of the anti-entropic—or negentropic—character of criticism is sometimes at odds with itself. From an information theory perspective, redundancy *lowers* entropy. More generally: the more predictable a text, the lower its entropy. Let me offer an analogy to illustrate this. Imagine Poet A who produces nothing but variations on Shakespeare’s 73rd sonnet—dithering the words here and there, but generally sticking to the form (e.g. “That time of day thou mayst in myself find...” or “That age of years you shall spy in me...”). Imagine now Poet B who writes each poem riffing off of the style of a different author—Rilke, Dickinson, Basho, etc.—never once repeating the object of imitation. Now imagine that each poet were to hand you their next poem enclosed in a manilla envelope. Even though you could see neither poem, you would have a better sense of what Poet A’s would look like. Poet B’s *oeuvre* is higher in entropy, more variable.

To please Flusser’s automatic critic that values surprise above all else, a writer would simply need to use letters (such as “X”), or words, or syntactic patterns that are relatively rare—taking a “model” and adding “noise.” But filtering based on such a criterion would make it so that the texts that would pass its critical filter would, as a corpus, be more entropic than those that had been filtered out; at the extreme, each text that passes the filter could be *sui generis*, absolutely unique and “improbable.” This literature might end up looking a lot like Poet B’s *Collected Works*, each of poems totally unrelated to all the rest. In other words, what Flusser seems to be describing is an automatic method for *accelerating* entropy, not pumping its brakes.

My point is not to quibble with the ways that Flusser interprets the entropy of thermodynamics or the entropy of information theory; these are notoriously slippery and even paradoxical concepts (Hayles 1990), not to mention ones that current science is still revising. Rather I aim merely to observe that the analogy between human editors and Flusser's demon starts to break down. After all, human editors—even of the most modernist variety—do not simply choose texts that are most “unique.” Individual poetry journals often possess distinctive styles. Even if the editor is not able to explain why (i.e. to formalize it in a way that could be turned into an algorithm), they may have a sense that a perfectly new-seeming poem does not fit their journal's ethos. Likewise, writing poetry is, for all but the most fanatically self-conscious “experimental” poets, something other than coming up with the poem that is most unlike all other poems that came before it. A poet might instead stick within a well-worn genre or form (say, the sonnet), trying to do something only slightly new with it. Forms themselves are generally anti-entropic; if the first line of a sonnet ends with the word “hide,” you might guess that the third line will end with “chide” or “debride” but not with the vast number of non-rhyming words. And certainly poets themselves tend to develop their own styles. If they did not, it would be impossible to speak of what is “Ashberian” about various examples of John Ashbery's verse. Poetry may be the art-form that is most concerned with the exploratory task of figuring out all the different ways that words can mean. But neither Erato nor Caliope are monomaniacs. Poets (and their editors) often have all sorts of goals, many of them more conservative. A poet, while trying to land upon some never-before-seen combination of words, may also try to express a quite familiar feeling such as fleeting joy or interminable longing.

Flusser's automatic critic is an invitation to think in post-human direction, imagining a way of judging writing that flirts with human-seeming judgment (i.e. aversion to cliché) but then, by taking it too seriously, makes a numerical idol of it. Modernists may have wanted to “Make it New”, but deciding whether language *is* new (whether it teems with unpredictability and “information”) is a job for a computer, not a human.

The Role of the Human

What role would humans have in a world in which human judgment could be replaced by the automated judgments of machines? Flusser begins by considering one possibility: that humans will likewise outsource creativity to machines and will themselves only design algorithmic “filters” that allow certain machinic utterances to pass into the world while others, like so much chaff, are discarded

(2011b: 120). Here Flusser could be describing the “objective functions” that designers of machine-learning systems provide to their algorithm. For instance, someone designing a system to generate a painting might demand that the computer maximize the similarities to Picasso’s Blue Period but also to minimize the presence of recognizable faces and objects in the painting. These competing objective functions, we might hypothesize, could spur the computer to create paintings that evoke Picasso’s but that are non-representational in nature. The computer itself would do both the creating and the critiquing; the human’s role would merely be to figure out how to translate some desired output (e.g. “like Picasso’s Blue Period but non-representational”) into specific, algorithmically-specified criteria.

But perhaps the human is not necessary even for this abstract responsibility. As Flusser worries, if algorithms can choose (i.e. be automatic critics), they can also choose their criteria for choosing. The human’s true freedom, their last and only decision, will be the power of “veto,” the “constant possibility of rejecting” this system itself. What Flusser predicts is nothing less than the elevation of human consciousness to a maximal level of meta-cognition: “telematics,” the algorithmic regime that thinks and creates in lieu of us, is “technology of freedom” because it “steadily broadens our view of the fundamental freedom to reject telematics itself” (2011b: 122).

Flusser is not wrong that the development of algorithmic media often does little by little obviate human values or displace them to ever-higher levels of abstraction. To support Flusser’s prediction, we might offer up the example of neural network models such as GPT-2 (Radford 2019). While these types of unsupervised models are trained to maximize a very simple (or abstract) objective function – predict the next word in a sequence given the previous words – their vast number of parameters trained against oceans of textual data means that they subtly intuit all sorts of criteria in order to hone their predictions. The human does not have to describe what a good sentence is, or a good paragraph, or a good poem; these aspects the model figures out on its own, refining its creative and critical capacity simultaneously.

However, I suggest that Flusser’s breathless anticipation of the end of culture – at which point humans will be nothing but “arbiters” of computational processes (2011b: 122) – skips over an earlier step that is implied by his teleological vision: humans may design automatic critics while also suffering the critical judgments of the selfsame algorithms. In fact, this is the situation in which we increasingly find ourselves. For instance, social media companies often use algorithms to filter our feeds. These filters either spread or muffle our posts based on how likely they are to create user engagement, but pursuing the objective function of maximizing attention can have catastrophic effects

on social relations and cognition. These algorithms can push us into “filter bubbles” (Pariser 2011), alienating us while distorting our perspectives. As Bernard Stiegler (2010) argued, the capitalistic “programming industries” such as Facebook are fundamentally incapable of truly caring for those many billions of minds that they have captured. Their demons are not as high-minded as the ones Flusser imagines.

Flusser’s vision of a world of fully-automated cultural production is radically egalitarian, almost a post-human communism: every person will be outside the creative-computer/critical-computer loop (and thus “useless” and free). In our world, however, power over computational media is far from equally distributed. The “programming industries” have had the opportunity to program and let loose their demons that filter our words and minds in ever more invasive ways. Individuals do not usually have this power. We do not get to control and modify our own newsfeed algorithms any more than we get to control and modify our own nuclear power plants, satellites, or transatlantic cables. We are everywhere “below the API,” but the API belongs to someone else, and we often cannot even inspect the code that critiques and judges our actions and our words.

But this need not be the case. In the rest of this paper, I illustrate the development of a piece of software that filters my own discourse, a small-scale “Flusser’s demon” that makes decisions about my own writing. I present this software not as a solution to a problem but rather, in the vein of “critical making” (Ratto 2011), as an attempt to think through the question of what kinds of “freedom,” what kinds of power, can be achieved by externalizing and automating one’s own critical criteria.

Designing StyleVise

StyleVise (or, where emoji are readily available, StyleVise) is a computational writing interface that challenges me to be more creative. It is one example of what I call (after the classical techniques of rhetorical training) “digital progymnasmata,” writing interfaces that challenge the minds of writers just as weights challenge the bodies of athletes (Booten 2019b).

StyleVise judges a human’s writing “without introducing any criterion other than rarity.” (Flusser, 2011b: 118) Earlier examples of my digital progymnasmata would scold the writer for using a particularly low-information word—for instance, describing the moon as “blue” or “white” (rather

than something unexpected such as “senescent” or “aquiline”).⁵ StyleVise applies the principle of filtering out rarity not to the level of lexis (word choice) but to that of syntax, by which I simply mean the structure of each phrase or sentence as opposed to the actual words that comprise it. In carrying out research on digital progymnasmata, it made sense to start with words, the most simple and obvious units of language. Syntax is not just more complicated but more mysterious.

When I decode a word, I noticed all there is to notice about it: “F” followed by “I” followed by “u” followed by “s,” and so on. Given two words, “Flusser” and “flintlock,” I can easily tell them apart. But the structure of a sentence is latent; before I can read the syntax of a sentence, its words must be first translated into abstract categories such as parts-of-speech and perhaps even varieties of phrases and clauses or more complicated structures such as the branching dendrites of a Context-Free Grammar. Syntax is not itself legible on the surface. It requires some further annotation. Given three words with very different rarities – e.g. “song,” “melisma,” and “harmony” – it is often not too difficult to sort them from most to least rare using nothing other than one’s own intuitions (one’s internalized model of lexical statistics). Because it is difficult to even conceptualize syntax, it is likewise more challenging to look at a random sentence and estimate its syntactic rarity. At this point, several thousand words into this short paper, I have some sense of what words I have tended to use more than I have in other prose I have recently written – certainly “Flusser,” “automatic,” “critic,” and “poetry.” I also have a sense of some of the quite rare words that I have used, such as “equi-page.” But I have no sense of what kinds of sentence structures characterize this piece. Have I repeated any syntactic patterns? Do I have “tics” that I myself do not notice? Are my sentences longer or more complex in this paper than in other examples of my academic writing? Is my syntax rarer or more common than other papers published in *Flusser Studies*? I have no idea.

I made StyleVise to address my sneaking suspicion that I do tend to overuse certain syntactic patterns. (No doubt I also overuse certain words, but, as I have argued, these are easier to self-diagnose.) StyleVise begins its work by taking over the responsibility of reading the syntax of the writer’s writing. It represents syntax in perhaps the most straightforward way possible: as the sequence of all of a sentence’s part-of-speech tags. So, for instance, the sentence:

O latest born and loveliest vision Of all Olympus’ faded hierarchy!

⁵ Boring (noun,adjective) pairs were extracted from the Project Gutenberg corpus, oft-used in both computational linguistic and computational poetic projects.

Can be “tokenized” into the following sequence:

```
["O", "latest", "born", "and", "loveliest", "vision", "Of",
 "all", "Olympus", "'", "faded", "hierarchy", "!"]
```

which can in turn be automatically assigned part-of-speech (“POS”) tags:

```
["UH", "JJS", "VBN", "CC", "JJ", "NN", "IN", "DT", "NNP",
 "POS", "VBD", "NN", "."]
```

This sequence, according to the conventions of the part-of-speech tagger I used⁶, corresponds to an interjection, a superlative adjective, a past-participle, a coordinating conjunction, a singular noun, a preposition, a determiner, a proper noun, a possessive ending, a past tense verb, a singular noun, and a terminal punctuation mark.

To create a *model* of syntax, I tokenized and part-of-speech tagged a large number of sentences—over two million in all—gathered from literary sources (Project Gutenberg texts). To balance out any stylistic idiosyncrasies of that collection of rather antique literary works, I added a similar amount of text drawn from contemporary, non-literary texts (Amazon reviews). Looking at sequences of these part-of-speech tags limited to a maximum length of seven⁷, I then simply calculated the frequency of each syntax pattern. The resulting data—each syntax pattern found at least once in my large corpus of texts paired with the number of times it occurred—is a simple “model” of how writers tend to organize their sentences on a syntactic level.

The game of StyleVise is played like this: my model of syntax patterns is divided into five groups, from the most common to the most rare. The game’s (intentionally rudimentary) interface challenges the writer to compose sentences. Each time the writer enters a sentence, StyleVise automatically part-of-speech tags it and compares it to a list of “banned” syntax patterns. At first these are the first group, the most common syntax patterns. Every so often, however, the game gets harder; more syntax patterns are included in the list of forbidden ones. It becomes necessary to avoid not just the most common patterns but also less and less common ones—hence StyleVise is a “vise” as

⁶ StyleVise uses spaCy, a popular collection of software for machine-reading and machine-writing (Honnibal and Montani). The conventions themselves are those of the Penn Treebank Tagset (Marcus, Santorini, and Marcinkiewicz 1993).

⁷ To be more precise: StyleVise looks at the syntactic n-grams (n=7) derived from part-of-speech sequences. This limitation is intended to more fairly evaluate sentences regardless of their length. A sentence may be syntactically rare simply by virtue of being extremely long, yet it may be composed of rather mundane sequences.

in “clamp.” Ratcheting up the difficulty still further, the program forbids the writer from repeating any successful sentence’s syntax pattern.⁸ StyleVise is also a “vise” as in an “advisor”; it points out the specific sequence or sequences of part-of-speech tags that it deems too common by repeating the user’s sentence with the offending words struck through (see Figure 1). The program keeps track of the writer’s score (the number of valid sentences), and it is possible to lose by using too many frequently-used syntax patterns.

I originally imagined StyleVise as something a writer might use every few days in order to become syntactically limber—though it would require scientific investigation to determine whether training with the interface would lead writers to use more invigorated syntax when not using it. From the vantage point of Flusser’s writing on the automatic critic, however, it is unimportant whether or not the human’s internal language model is enlarged. What matters is merely that the assemblage of the human and the machine collaborate to inject ever-noisier noise into language, a system that is not the exclusive domain of humans.

Playing not One Game but Two

Writing with StyleVise is an anxious experience, not simply because its advice comes in a rather scolding tone but also because pleasing its whims demands a grotesquely positivist way of relating to language. How do I know that this sentence is rare enough? Have I used it before? If I am berated, what is the easiest way to edit a sentence to please the system? Playing it, I feel the temptation to win at all costs. Even a small and private failure represents some blow to the ego, however glancing or weak. Yet “beating” StyleVise is actually quite easy. It requires nothing more than coming up with the most random sentences possible with absolutely no regard for whether or not they are meaningful. (Four prepositions in a row? Why not!) It is well known in computer science that machine-learning algorithms, when tasked with maximizing some objective function, will sometimes behave in perverse ways to do so, evading the programmer’s intent while satisfying “the letter of the law.” When I play StyleVise only to win, I let its objective function lead me into meaninglessness. Like running a marathon by secretly catching a taxi to the finish line, this activity becomes a waste of time.

⁸ The command-line version also keeps a permanent record of the user’s, so each time she starts the game it is harder than the last; this is currently not possible with the web-based version.

Another programmer with more of an engineer's habitus might observe that my objective function is simply too simple. I need two objective functions: the writer should be forced to maximize syntactic rarity but without slipping below some minimal level of textual coherence, without resorting to writing sentences that are rare because they are gibberish. Encoding "not nonsense" as an objective function is more difficult than encoding syntactic rarity, but it is far from impossible. I might, for instance, train another syntactic model based on randomly-shuffled part-of-speech sequences. StyleVise could demand that my sentences are more like real sentences than these fabricated specimens. Yet this approach—adding objective functions until the writer cannot escape writing meaningful sentences—would represent a misunderstanding of the advantage of algorithmic critics and censors.

StyleVise is not a game; it is *half* of a game, one that actually consists of two games. The other game is not digital. It occurs in the mind of the poet—my mind, I suppose, since I am the primary (and perhaps only) user of StyleVise. The point of this second game, which occurs simultaneously with the other part, is to write a "good" poem. What constitutes a "good" poem cannot be totally specified (i.e. made algorithmic) beforehand, in part because it develops as I write. Perhaps these two games are in opposition. On the other hand, perhaps playing the first game—the game of pleasing the algorithm—actually makes it possible to play the second half with more sensitivity, panache, or joy.

Poets should be especially adept at juggling these two games, since they have historically subjected themselves to all sorts of numerical restrictions (14 lines in a sonnet, 2 syllables [1 long, 2 short] in a dactyl, 5 iambic feet in a line of blank verse, 2 ending rhymes in the entirety of a villanelle). They have always had to play the two games at once, playing them together as one—the first game of writing a "valid" poem (where the numbers add up) and the second game of doing far more than that, writing a poem that models some aspect of experience in ways that seem helpful, true, revelatory, etc. In other words, poets in particular have been training for this moment when the secret numericity of language, the statistical heart of it that was once obscured, now emerges to the fore. Following Flusser's liberal redefinition of "poetry" as "the computation of existing models with intrusive noise" (2011a: 73), we may add that poetry is the art of turning numbers back into numinous breath.

With anxiety and even sorrow, Flusser realizes that computation "desanctifies" poetry by revealing language as (like everything) susceptible to logical manipulation and brute calculation (2011a: 75). This realization will become more widespread as machine-learning systems improve, though

already they *really can* discover and mimic latent statistical realities that characterize the exalted discourse known as “poetry” (which, like any other discourse, is merely a sequence of words or letters). Linguistic positivism’s power cannot be denied, and the algorithms in its service are only getting stronger.

The question, then, is how to make a beginning of the end. Cast down from Mt. Helicon, how do we make this descent a *felix culpa*? Perhaps, as Flusser speculates in *Does Writing Have a Future?* and *Into the Universe of Technical Images*, all writing of poetry outsourced to machines who are better at adding “noise” than we are anyway. Another future is the one that keeps the human in the loop to play the computer’s game but also another one, to learn from what the algorithm has to offer while also rebelling against its form of knowing by offering something else, something better. Countering the precision of algorithmic media, poets should cultivate desires that are flighty, mercurial, intuitive, and (above all) beyond their own powers of articulation. In other words, poets will need to become “wise as serpents and innocent as doves.”

Scaling Up

Papers describing a new piece of software customarily conclude by suggesting how this work of engineering might be further developed—e.g, which problems could be addressed, which features added. In the context of writing on Flusser, a theorist whose work concerns not the future of technology but of the human spirit amidst technology, I feel instead the need to imagine what social futures might emerge should demons like StyleVise proliferate through our technical universe.

Clearly, moving towards the future of fully-automated criticism that Flusser imagines would require algorithmic sensors with more *mobility*. At present, StyleVise lives in two places: on my computer (where I run the command-line program) as well as online, at a specific website, as a web-based demo. (Its raw code also resides in a Github repository.⁹) Software becomes “universal” when it begins to appear more places and when its presence becomes less and less noteworthy. I am not surprised in the least to see spell-checking software living in my copy of Microsoft Word but also my email program, my social media interfaces, my banking website, and essentially all other locations into which I type text.

⁹ <https://github.com/kbooten/stylevise>

More and more of our typing is public, taking place on social networks or other platforms for immediate publication and sharing. Social networks would make a more-than-suitable habitat for StyleVise and other Flusserian demons. Imagine if the engineers of the social network Twitter were to deploy to the site a demon whose job it was to check every new tweet against all those billions that have already appeared on the site, filtering out any that have already appeared, or banning users for writing too many tweets that are similar to others in the archive. Since they own unrivaled quantities of user-generated text, social media companies are particularly well-positioned to statistically determine what expressions are common and which are rare. And, since social media companies tend to reserve the power to interfere in their sites, they could at any time delete or boost posts based on the demonic appraisals of their algorithms. In fact, social media companies often already use bots to remove users' posts for possible misinformation or for offensive content. Flusser's vision of a telematic future is much closer than it may at first appear.

On this Flusserian social network, monotony and redundancy would be banished. If the avid users of social media users keep scrolling in order to find something *new*, the network would cater to this instinct even more thoroughly. Yet the results could be catastrophic. When societies place too much emphasis on a single metric (e.g. GPA), they can slip into a state of *involution*—a “a highly dynamic trap” in which each individual must constantly fight to get ahead merely in order to stay at a current level, since their peers are also scrapping to outstrip them (Xiang Biao, quoted in Qianni and Shifan 2020). As Xiang Biao has seen in the context of contemporary China, involution leads to anomie, since each person must regard all others as competitors in a zero-sum game. Let loose on Twitter or Facebook, the Flusserian demon would likewise accelerate and perfect the involutory character at the heart of social media. On Twitter, posters must constantly generate engaging content in order to acquire followers and earn plaudits (“likes” and “retweets”), but at least now there is a dignity to being somewhat unpopular, since it is possible to rationalize one's low-follower account as simply “niche.” Such illusions would be impossible on a Flusserian social network, since all users would be forced to conform to the same objective function. Each time a post would pass its informatic filter, the task of posting something new would become more difficult; as with StyleVise, the linguistic constraint would gradually ratchet up. A Flusserian social network would also quickly become a zone of ever-increasing alienation, since forging affinity or political solidarity is predicated upon groups of people saying the same things—the same opinions, the same argot, the same slogans and mantras. Longing for connection and sick of mandatory individuality, individuals might figure out how to trick the demon, saying the same thing as each other using words that seem just different

enough to evade filtration. But here again Flusser's pessimistic sense that all human expression is reducible to algorithmic logic seems to hold. The engineers of the Flusserian social network would gather up such posts and use them to train the algorithm to detect such human deception.

But StyleVise, I object, is not for the "universe" but for me. Indeed, I made it for myself and myself alone, though others may download and use it. I wanted it to help myself to write beyond what I would ordinarily write, to goad myself to step over the hedges of my imagination that keep my expression confined. I *feel* myself walking over my own footsteps, beating the grass down into familiar paths. I wanted to be forced to leave the comfort of circumambulation and to trek and range. If StyleVise incentives exploration, it also opens up a new danger: using it every day, I would become trapped in an involutory contest with myself—and, should others use the software for some reason, with them. Underwriting StyleVise is an athletic metaphor, the one that echoes in the *gym* of *progymnasmata*. Perhaps by submitting to weeks or months of training with StyleVise, my syntax muscles would become stronger and more flexible than they are currently. I might even become more expressive than at least a few other poets, moving into a more rarified percentile of syntactic flexibility. But, if this is the case, no doubt I should not have written this paper. I should have instead kept StyleVise a secret, just as a painter might keep secret a camera lucida, just as a baseball player would keep secret his performance enhancing drugs. Involution happens when education is reduced to pure desire to "get an edge." One can get an edge in all sorts of ways, from staying up later than one's competitors to hiring a better tutor. Or one can pay for software such as Grammarly, an artificial critic to filter out all textual infelicities. The spell checker now offers no economic benefit, having become almost universal. While producing unexpected syntactic structures also likewise does not confer much of an economic benefit (and, since plain writing is more efficient, may even make it more difficult to score a better job), we may not be far away from a future of diverse Grammarlies—one to filter in only those statements that suggest a sparkling wit and erudition, another to filter out any statements that are not in keeping with the politics or sense of humor of the teacher or the boss. Only the wealthy will be available to afford them all; the ultra-wealthy might then pay machine-learning graduate students to create specialized filters for their striving progeny. Performance in the information economy may end up being a demonic arms race. I am beginning to think that I should not have made StyleVise at all, and that, if I even remotely ethical, I will delete its code from the web.

But perhaps some other version of this sort of human-assisting software, some other Flusserian demon, could lead me beyond the dull confines of my too-predictable mind but not into the hyper-future of endless competition. The sonnet is a kind of algorithm for the mind, one that must

be beneficial to our thinking or else we would not have clung to it so fiercely.¹⁰ The sonnet is also a filter. It filters out (keeping in the realm of unexpression) all poetic utterances that do not abide its required rhyme, meter, and rhetorical structure. But these are only first-order characteristics: by following the form's instructions we write something better than we could have written without them. The paradox of form is that, by agreeing to give up certain degrees of freedom, writers gain a chance at breaking out of their familiar patterns, their own exhausted forms. Yes, one may write sonnets competitively, seeking to master this form or make it one's own, but this is only incidental. The gambit of the poetic form is that its demands are not arbitrary but rather inherently salubrious.

I began this paper by arguing that practitioners of electronic literature should dedicate themselves to designing automatic critics, algorithms that will filter human language. Now it is possible to make this task more specific: what we need is the automatic critic equivalent of a sonnet, a Flusserian filter that, like the sonnet, generates beauty but is also itself beautiful. And one that is, like any form worth mentioning, a public good.

¹⁰ For a further discussion of this point in light the implications of “cognitive poetics” for designers of electronic literary media, see Booten (2020), in some ways a companion piece to this paper.

References

- Booten, K. (2019a). Harvesting Rerites. In: Johnston, D. J. *ReRites: Human + A.I. Poetry + Raw Output + Responses*. Montréal, QC: Anteism.
- Booten, K. (2019b). Toward Digital Progymnasmata. *Proceedings of the 10th International Conference on Computational Creativity*: 1-8.
- Booten, K. (2020). Making Writing Harder: Computer-Mediated Authorship and the Problem of Care. *Proceedings of the Electronic Literature Organization Conference and Media Festival*.
- Flusser, V. (2011a). *Does Writing Have a Future?* Nancy Ann Roth, trans., Minneapolis, MN: University of Minnesota Press.
- Flusser, V. (2011b). *Into the Universe of Technical Images*. Nancy Ann Roth, trans., Minneapolis, MN: University of Minnesota Press.
- Goodwin, C. (2018). A.I. Storytelling: On Ross Goodwin's 1 the Road. *BOMB: bombmagazine.org/articles/ross-goodwins-1-the-road/* Accessed on October 1, 2020.
- Hayles, N. K. (1990). *Chaos Bound: Orderly Disorder in Contemporary Literature and Science*. Ithaca, NY: Cornell University Press.
- Honnibal, M., Montani, I. & contributors. (n.d.). spaCy. github.com/explosion/spaCy
- Lutz, T. (1959). Stochastische Texte. *augenblicke*, 4: 3-9.
- Marcus, M., Santorini, B. & Marcinkiewicz, M. A. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2): 313-330.
- Pariser, E. (2011). *The Filter Bubble: How the New Personalized Web is Changing What We Read and How We Think*. New York, NY: Penguin.
- Project Gutenberg*. [gutenberg.org](https://www.gutenberg.org). Accessed on October 1, 2020.
- Qianni, W. & Shifan, G. (2020). How One Obscure Word Captures Urban China's Unhappiness: Anthropologist Xiang Biao Explains Why the Academic Concept of "Involution" Became a Social Media Buzzword. *Sixth Tone*. [sixthtone.com/news/1006391/how-one-obscure-word-captures-urban-chinas-unhappiness](https://www.sixthtone.com/news/1006391/how-one-obscure-word-captures-urban-chinas-unhappiness) Accessed on November 10, 2020.
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. & Sutskever, I. (2019). *Language Models are Unsupervised Multitask Learners*. https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf Accessed on November 10, 2020.
- Ratto, M. (2011). Ratto, Matt. Critical Making: Conceptual and Material Studies in Technology and Social Life. *The Information Society*, 27(4): 252-260.
- Sloan, R. (2016). Writing with the Machine. *robinsloan.com*: robinsloan.com/notes/writing-with-the-machine/ Accessed on October 1, 2020.

- Stiegler, B. (2010). *Taking Care of Youth and the Generations*. Stephen Barker, trans., Palo Alto, CA: Stanford University Press.
- Strachey, C. (1954). The “Thinking” Machine. *Encounter*, 3: 25-31.
- Rao, V. (2017). The Premium Mediocre Life of Maya Millennial. In: *ribbonfarm*: ribbonfarm.com/2017/08/17/the-premium-mediocre-life-of-maya-millennial/ Accessed on October 1, 2020.
- Wark, M. (2015). The Vectoralist Class. *e-flux journal*, 65: supercommunity.e-flux.com/texts/the-vectoralist-class/ Accessed on October 1, 2020.

Figures

```

StyleVise
Type "play" to start a new game.
Type "help" or "?" to list commands.

>play
*****
Level 4: EXPERT
*****
    Type 'score' to see your current points (👇) and your remaining life (❤️).
    See how many points you can get before you run out of life!

>The light came through the window.
    The light came through the window -
    ...ZZZZZ...
    ❤️ -1

>The heavy light came through the window.
    The heavy light came through the window -
    Is that all you got?
    ❤️ -1

>Light moved again through the window.
    Light moved again through the window -
    ...ZZZZZ...
    ❤️ -1

>The light of the air became the light of the window.
    The light of the air became the light of the window -
    ...ZZZZZ...
    ❤️ -1

>The light whose home was air became the light of the window.
    The light whose home was air became the light of the window -
    Same old same old.
    ❤️ -1

>The light whose home was air becomes now the window's delight.
    The light whose home was air becomes now the window 's delight -
    *Yawn*
    ❤️ -1

>The light whose home was air becomes now the light whose home is window.
    👇 +1
  
```

Figure 1: Writing with StyleVise. (Command-line interface shown here.)