Import This

Buffy Cain

What draws me to conceptual writing is a certain kind of playfulness that has as much to do with board-games as with Derrida: desire to experiment with the effects of building or manipulating forms, and through this process of experimentation determine the conditions of a system, or condition the determination of a system. Conceptual writing renders a conceptual model — 'renders' as in "rendering is the process of generating an image from a 2D or 3D model," 'conceptual' as in "Alison Gopnik is a pioneer of concept-learning in developmental psych", and 'model' as in "models are structures that can be used to represent or reconstruct phenomena" — by iteratively treating its own sentences as objects in the formal system of a text, then treating texts as objects in the formal system of language, then treating languages as objects in the formal system of a culture, climbing up by overworking the capacity of each subsystem to reflect its meta-system til it generates a crude but totalizing (so grotesque and/or revelatory) model of the meta-system one degree above, then repeating with the model (of the meta-system—now a sub-system) as the input.

Maybe I should double back and clarify what I really mean when I say 'system'. Systems, as conceived of in the context of cybernetics, are sets of interacting components which form coherent wholes. They exhibit structure in that they are composed of interconnected parts (which are themselves sub-systems), and express purpose through their function, and this structure and function is more definite than that of the surrounding environment[1]. Systems can be abstract or physical, living or lifeless, open or closed. Language can be conceived of as a formal abstract system, in that texts are their own kind of *things* and language is a kind of mathematical and computational *structure* of such *things*. Language is also a constituent part of various encompassing super-systems, e.g. the conceptual model (a kind of knowledge structure defined in cognitive science and in computer science) which can be instantiated in information systems, in the human mind, and in works of art or literature. Okay.

To motivate discussion of how it is that conceptual writing renders systems, I'll sketch the contours of two works of conceptual writing and the systems they render.

Michael Anzuoni's 'Anyone Within a 5 Mile Radius Who is Single and Smokes' is constructed from the output of a bot named Bot Chuck_Kinbote, a Markov process trained on text scraped from online dating profiles on OkCupid. A Markov process generates text based on a stochastic model of possible suffixes for a given prefix in a corpus (the data structure is a directed graph with a node for each word in a text and weighted links between nodes)[2]. Anzuoni then sent the model's output to anyone on OkCupid

---

1 . Backlund, Alexander (2000). "The definition of system". In: *Kybernetes* Vol. 29 nr. 4, pp. 444–451.

2 Everitt,B.S. (2002) *The Cambridge Dictionary of Statistics*.

within a 5 mile radius of his location who was single and smoked. The final conceptual work is a presentation of the bot's activities, and it has a pleasant if slightly absurd surface reading:

> *It's a/Reddit/Job/My waterbottle/Shouting along engaging in 7th grade/Oxygen (I like chipotle/What a/Taking a lot/With a/Getting a camera, and watching Netflix, wasting time hanging out at home early to work when I suppose you again/and should throw all nit and not)*

'Anyone' explores the OkCupid platform (the overlay of a social system on a digital infrastructure) by creating an exaggerated masculine agent, hyperactive to the point of frantic aggression, and releasing it into the ecosystem. The project takes the shotgun approach and the imitative gesture ('Look how much we have in common!!!') and accelerates and intensifies these stock tactics, creating a maybe too-literal 'desiring machine'. The desiring machine[3] acts as a component of a larger social machine, and is a consuming/producing force which appropriates whatever is external to it in order to constitute itself and participate in the construction of social realities. By adhering to the constraints and exploiting the facilities (adhering to the structure and the function) afforded by the OkCupid platform, 'Anyone' reveals the aggressive masculine relation to the other as an immanent product of the network, a passively emerging result of the structured systems through which desire is channeled.

Funny story: while Chuck_Kinbote was manically regurgitating text, I also had my own Markov Chain deployed on Okcupid. Ur-buff was trained on my emails and chat transcripts. She was passive (never sending the first message), accommodating, and reliably produced self-disclosures. Sadly, Bot Chuck_Kinbote and ur-Buff never made contact despite geographic proximity, because I was lying to myself/everyone else about that fact that I smoke (I'd like to think that in another world-line our bots met and found love, or well…company, at least). Much like Bot Chuck_Kinbote, ur-buff inadvertently lands in the realm of exaggerated behavior, her output bordering on hysterical clinginess:

> *I don't know I don't know…I would be lovely to go out with meeee. Ok I'll let me terrifying the first choice for your way I think… come home for my way back when to see how I want to comeee dancing around till like you want / it's fine. You're not a problem for the first choice grass for my sweet and the seven years. Let's get the Nassau. I love you. I'm in Harlem... Come back.*

I undertook this not as an artistic project, but as a (formerly) private means of coping with the inherent physical and psychological trauma of dating. The rules for performing femininity in this online pseudonymous space were fundamentally excruciating, so I automated and accelerated the functions I found myself performing. (Let's keep it chill and context-sensitive about the gendering of formal gestures though: this year I'm writing a The Waste Land/"A"/The Cantos type of poem-about-all-of-poetry by mathing 'all of poetry' into a bot so I can write through all of poetry efficiently by thinking through the songs that my mechanical All-Poetry-canary sings when I command it to.)

The second work I'd like to address is Trisha Low's 'The Compleat Purge', which deftly assembles a 'regurgitation of tropes and societal models' as mediated through blogs, fandoms, and chat rooms. Finally 'out' as a structuralist (I mean we all kinda suspected, right?), Low states that she 'likes to play

---

[3] Deleuze, Gilles and Félix Guattari. 1972. Anti-Œdipus. Trans. Robert Hurley, Mark Seem and Helen R. Lane

with blocks'[4], and the work she creates by assembling these 'blocks' of text has an architectural quality to it. A series of Last Wills and Testaments are placed alongside chat transcripts of female fans role-playing sexual fantasies as male pop-stars, a eighteenth century didactic romance novel, and an incisive—there's a pun here—essay about authenticity and conceptualism. Low depicts self-disclosure, circumscribed desire, and pressurized excess—the teen-girl Tumblr aesthetic—as attributes that emerge from the oppressive or authoritative structures she lives within, and treats texts as interconnected modular functions and feedback loops capable of almost diagrammatically illustrating those processes that produce the Tumblr aesthetic. For instance, in the first section of 'Purge', she places a series of Last Wills and Testaments (unconsummated suicide notes) back to back, demonstrating how the affects and attachments in each one intensify or diminish as they feed into the next, thereby illustrating the feedback loops at work in her psychosocial reality. Low has also stated that she wants to twist form via increased 'pitchiness', not through subversion or polemic but via 'repetition, but with increased intensity and compression—a torque'[5]. The feature that allows us to pick out a system from its surroundings is just this kind of 'torque', this more definite structure and behavior of the system relative to its surroundings.

Low's and Anzuoni's works are constructed according to a similar logic—a data structure is transformed algorithmically into a text which models a system in a system in a system. (Algorithms, by the way, are not by definition digital. The mathematician Al-Khwārizmī coined the concept of an algorithm in the 800s to mean any formalization of a protocol as a procedure.) Anzuoni's 'Anyone', literally algorithmic, models OkCupid with statistical computer code. Low's 'Purge', algorithmic in its loyalty to an intractable set of internal rules you know are there because you *feel* the 'legible and restrictive conceptual form'[6] flattening the raw material of the text, forcing the reader to perceive the social and psychological systems acting in/on the mess that is our feelings, makes the system 'Trisha Low' into a model of the system Tumblr. And what is metaphorically algorithmic in Low's thinking processes is also, let's remember, literally algorithmic: electricity moving through biological circuits doing computations. So both of these conceptual works are a product of the symbiotic relationship between the algorithm and data structure, the user and the system. They are a projection of the ontology of the digital onto the surface of culture, the implosion of an information system into a text, into a mind.

Generally speaking, the processes by which conceptual work is produced—repetition, sorting, replication, constraint, permutation, stochastic or methodical sampling, recursive embedding, compression/decompression, and distortion—operate with a similarly algorithmic logic. If conceptual works are built by algorithms acting on data structures, then it makes sense that the process of making and distributing this work has become increasingly entwined with the digital, as the digital has particular facility in representing and enacting data structures and algorithms, and is itself comprised of them. The digital affords us features that are productive in generating conceptual work: technological implementations for protocol, modularity, and abstraction. Protocol gives us functions which can serve

[4]Low, Trisha. 2014. 'Trisha Low by Sarah Gerard' BOMB magazine.

[5]Low, Trisha. 2014. 'Trisha Low by Sarah Gerard' BOMB magazine.

[6]Low, Trisha. 2013. 'The Compleat Purge'

our purposes, modularity allows for the organization and reorganization of encapsulated sub-systems within a system, and abstraction allows for the management of complexity (To do a callback: systems "exhibit structure in that they are composed of interconnected parts (which are themselves sub-systems), and express purpose through their function"). Conceptualism use of protocol, modularity, and abstraction on the quest for new forms of coherence stages the negotiation of the subject—already a networked and linguistic being, and so itself a system — with the world as an absurd game. The whole endeavor's playful, in the sense of playing tabletop games which use rules and tokens to allow a group to engage in modeling a complex or abstract system in an always slightly fucked up way.

Protocol: The mathematical and computational structures of language can be viewed as defining possible protocols that can implemented as functions. Functions, in computer science, are a means of formalizing the process of accessing an object's data-structure so that modifications might be performed efficiently. A familiar example of a linguistic protocol capable of being implemented as function is alphabetical order, which allows a collection of words to be structured as a totally ordered set, rendering it easily navigable by someone privy to the ordering convention. Tauba Auerbach's 'Alphabetized Bible' explores the result of re-organizing the bible according to the principle of alphabetic order, trading the complex and contradictory tenets of the bible for an explicit and simple logic (fun fact: both alphabetical order and the bible originate in the same region and time period[7]). Another familiar protocol, especially in the context of lyric poetry, is phonetic structure, from which alliteration, rhyme, and consonance arise. in Steve McLaughlin's 'Puniverse', the permutative potential of rhyme is used as a function to expand a finite (but enormous) set of idioms to amusing and then exhausting effect "a brattle of wits/a cattle of wits/a chattel of wits/a prattle of wits/a rattle of wits/a tattle of wits". At the level of syntactic structure the protocols for recursion, embedding, and combination come into play. The first section of Nick Montfort's Megawatt explores the combinatory logics of embedding in Beckett's Watt, extending the original text of Watt by at least an order of magnitude by adding three new verbs and then following Beckett's combinatorial lead.

Modularity: Many conceptual texts are less uniform than the examples that I've just given, and are instead assemblages comprised of multiple heterogeneous texts. Modularity allows the writer to string together a series of functions or structures in order to represent a more complex system. In programming practice, object and function composition allow for modularity. Object composition facilitates the formation of complex data structures, such as the tree or graph, by specifying relationships (e.g. a dog has a tail, a bag contains chips), and function composition (a modular use of protocols) enables the creation of functions with greater complexity by allowing the output of one function to flow into the next. I've alluded to modularity in Low's work already. Tan Lin and Cecilia Corrigan also build work in a modular fashion, assembling appropriated texts, found texts, texts fully determined by genre, or generated/sampled text in structured relation to one another in order to instantiate feedback loops, exchanges, filters, interruptions, ruptures, and jumps. The controlled vocabularies of Lin's 7 Controlled Vocabularies ostensibly structure reading in this 'ambient

[7] Lehmann, Reinhard G. "27-30-22-26. How Many Letters Needs an Alphabet? The Case of Semitic", in: The idea of writing: Writing across borders / edited by Alex de Voogt and Joachim Friedrich Quack, p. 11-52

autobiography of a book about ambient autobiography', but inevitably the form distinctions buckle under the weight of the included meta-data (barcodes, receipts, labels, postcards, library classification codes) that 'point' outside of the assembled texts. The proliferation of meta-data constitutes an insistence that language is caught up in structures or mechanisms of exchange[8]. The textual flows from art-book excerpt, to index, to poem, to fiction, to theory, to autobiography begin to mirror the flows experienced when ambling through a metropolis, perhaps into a museum, then into bodega, onto the street, and back into a theatre or a library, reading signs and labels overhearing stories, collecting scraps, trinkets, information, and narratives in exchange for presence and attention.

Abstraction: Abstraction is a means of suppressing complexity in order to provide an idealized interface for the programmer to work within. In both programming and in conceptual writing abstraction serves to simplify the process of both building functions and manipulating structures. In programming language theory, the 'Abstraction Principle' states that each significant piece of functionality should be implemented only once in the source code, and that where similar functions are implemented by different pieces of code, the differences should be abstracted away so that they be combined into one function[9]. Abstraction thus works from the bottom-up to simplify the job of the coder, ensuring they don't duplicate work. Conceptual writers show an analogous reluctance to duplicate work, and readily sample or full-on appropriate texts to serve their needs. But abstraction flows the other way as well. Abstraction, as a top-down process, allows us to interact with complex systems by stripping away their detail and complexity, reducing them to their strongest structural and functional features. This allows us explore and exploit their functions, even if it's outside the scope of our abilities to know how exactly it is that they do what they do. Scientists (physicists, cognitive scientists, and economists are the worst offenders) rely on abstraction to construct models, and hope their analogies hold as their model's predictions are tested. And these models only ever kinda work. If cartography achieved perfect rigor, maps would be as big as their territory (Borges). Modeling is reductionist, but damn it sometimes you have to get abstract if you want to represent the complex. Sense isn't made without sacrifice. Conceptual writing is enamored both with the power of abstraction to create insight and with its power to flatten away everything of worth—best if you can do both at the same time, as Tan Lin never fails to.

Some ideas in this piece play off of dissertation work in progress by Peli Grietzer.

---

[8]Gallagher, Kristen. “Cooking a Book with Low Level Durational Energy: or, How to Read Tan Lin’s Seven Controlled Vocabularies”, in: Reading the Difficulties: Dialogues with Contemporary American Innovative Poetry edited by Thomas Fink, Judith Halden-Sullivan, p. 93-100

[9]Pierce, Benjamin (2002). Types and Programming Languages..p. 339