THE UNIVERSITY OF CHICAGO

READING ALGORITHMS

A DISSERTATION SUBMITTED TO

THE FACULTY OF THE DIVISION OF THE HUMANITIES

IN CANDIDACY FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

DEPARTMENT OF ENGLISH LANGUAGE AND LITERATURE

BY

DAVID MARTIN THIEME

CHICAGO, ILLINOIS

JUNE 2020

DEDICATION

I dedicate this to the memory of my mother, Ann Ruttledge, and my father, Allen Thieme.

Table of Contents

Table of Figures

Abstract

The current rise of Machine Learning (ML) and the proliferation of ML-based algorithms in modern technology has led to renewed speculation that Artificial Intelligence (AI) could soon match or exceed human cognitive capacity. The ability for an ML-based system to *learn*, combined with the increasing proficiency and capacity of "deep" ML algorithms lends credence to this speculation, and gives rise to imagined futures—some promising, some apocalyptic—in which machines can think like humans.

Many ML algorithms operate on linguistic-based data. Digital assistants such as Siri and Alexa cognize and enact users' commands, while Google returns extremely relevant datasets based on a few keywords. We increasingly read digital content and information generated by algorithms, such as the generated advertisements that appear in web browsers. ML algorithms are thus becoming increasingly pervasive and effective at reading *us* and determining what is best to read *for us*.

The sophistication of modern ML algorithms thus calls into question the boundary between algorithmic and human cognition, and the proliferation of ML in modern devices is increasingly forming a technological substrate that reads and writes us into the world at timescales below experiential perception. But as it stands now, even the most sophisticated ML algorithms cannot approach human proficiencies for general reading and writing. And yet, some ML algorithms can perform certain specialized forms of reading and writing remarkably well. At this point, I do not believe either the technical or humanistic communities have developed the necessary critical methodology to formulate *when* and *why* these algorithms fail

or succeed. And this is a crucial discussion to have both inside and outside of the academy. As ML grows *quantitively* more sophisticated and proficient, it becomes increasingly important to understand and articulate the *qualitative* gulf between human and algorithmic cognition. *Reading Algorithms* will thus attempt to describe a methodology that deploys the algorithms themselves as tools to demarcate the evolving boundary between qualitatively different modes of cognition. In the pages that follow, a humanistic understanding of interpretive reading will be deployed to highlight the qualitative differences in ML-based reading and writing. But crucially, we will also work in the opposite direction, taking a technical understanding of how ML algorithms consume and cognize textual data as an alternate "language" in which to formulate abstract literary-critical concepts.

These literary-critical concepts will be explored through the notions of *sequence, context,* and *attention*. First, *sequence* will be aligned with the literary-critical concept of narrative morphology, exploring how "algorithms that read" sequence data in a qualitatively different manner than human readers might arrange the formal features of text as an aesthetic experience. Second, in order to describe the complexity of modeling something as abstract as literary critical *context* within the contours of a machine, we will use technical knowledge of several algorithms to analogize what it might mean to "read like an algorithm." Finally, I will use the technical concept of *attention* as it functions within a complex ML algorithm to explain why "reading the output of an algorithm" as a literary text can open a collaborative space for humanists and technologists to understand the capacities and limitations of the underlying code. In these three conceptualizations of *Reading Algorithms*, I will demonstrate how both

humanistic and technological approaches to texts become necessary to understand different modalities of interpretation and cognition.

Introduction

Hey Siri?

Yes?

What is reading?

Reading is the complex cognitive process of decoding symbols to derive meaning. It is a form of language processing.

Hey Siri?

Yes?

Can you read?

We intelligent assistants are highly literate.

Hey Siri?

Yes?

Can you write?

I didn't get that.

Siri, the digital assistant software familiar to anyone who has used an iPhone, is an algorithm. Siri is instantiated in circuitry and code, but she (or he/they/it, depending on your settings and imagination) doesn't exist wholly within the contours of the physical space of your phone. Siri can tap into vast networks of information that are collated in Apple's datacenters across the planet as well as the distributed enormity of the internet. Siri can reach out to satellites in orbit to determine where she is; she can listen and speak. And, perhaps most importantly of all, Siri can learn.

To call Siri an "algorithm," then, seems almost a slight, as she seems like so much *more*. But Siri is still code, and her behavior conforms, exactly, to the instructions that were written by

the teams of programmers who created these extremely complicated components of software and hardware. And yet, Siri *learns*, and how and what this software learns is determined not only by the phone's owner—their voice, their location, their preferences—but also by events wholly extrinsic to the physical relationship of user and device. The constant flow of information from the internet, the presence of other devices, the backdrop of evanescent physical phenomena that may lay far beyond the reach of the phone's sensors but are nonetheless relayed to it via the vast reach of its network, all of these and much more contribute to the evolution of the algorithm.

Even in the face of this complexity Siri is still an algorithm – an unambiguous set of rules. How is it, then, that Siri can learn, adapt, and change its behavior? How can a rule-based system *evolve*, and furthermore evolve in the complicated ways that Siri demonstrates? Think of one of the first algorithms any schoolchild learns, arranging numbers vertically to aid in addition ("carry the one," etc.). This set of rules, when applied correctly, always yields a determined result. We are accustomed to associating algorithms with deterministic results. Even in our current era of computation, nearly all programs are deterministic: given a set of inputs and enough perseverance, one can always predict the output.

This common perception – that all algorithms are deterministic – has radically shifted very recently, and this shift is tied precisely to the adoption of various forms of Machine Learning (ML) and their proliferation in tools such as Siri. Siri still follows rules, and its behavior evolves according to those rules, but the results are, for all practical purposes, nondeterministic. That caveat—"for all practical purposes"—is crucial as it lies at the heart of why this investigation will be located within the humanities as opposed to the realms of

computer science or computational linguistics. For it is not quite true to call Siri—or any ML based algorithm—nondeterministic. It is conceivable that even the most complex forms of ML algorithms could be decomposed into millions (or billions) of individual computational steps, and each of those steps has a deterministic outcome that is described by the algorithm. But here is the key point: it is inconceivable that a human could cognitively grasp the entirety of a sufficiently complex ML algorithm *as a system*.[1] And this represents a major tectonic shift in the history of technology. We have historically understood tools as causal agents, extensions of our own will into the world. The tool and our interaction with the tool instantiates a chain of events in the external world, and if we can cognitively grasp each link of this chain as a cause of those that came before, it becomes much easier to understand the tool as *belonging* to the self-expressed-as-will. But complex ML algorithms introduce a rupture into this perceived continuity as we are so manifestly unable to cognitively grasp their emergent behavior in its entirety as a causal process. There is something fundamentally alien about this, an alterity that moves the tool away from our will and ourselves. For perhaps the first time in the short history of widespread consumer-based digital systems, even the most casual users of smart devices

[1] We will see many more references to ML systems, especially modern "deep" learning algorithms described as "magic," even within the technical community. A recent, non-technical article in *Forbes* notes this trend and echoes my claim above that these algorithms exceed humans' capacity to understand them as deterministic systems: Leetaru, Kalev. "Today's Deep Learning 'AI' Is Machine Learning Not Magic." *Forbes Magazine*, November 15, 2018. https://www.forbes.com/sites/kalevleetaru/2018/11/14/todays-deep-learning-ai-is-machine-learning-not-magic/. A more nuanced take on this opinion occurs in the *Harvard Business Review*, with the interesting addition that "augmented" ML tools are being created to explain what is behind the "magic," i.e., what data, in particular, is driving an ML system's decision making process: Abbasi, Ahmed, Faizan Ahmad, and Brent Kitchens. "The Risks of AutoML and How to Avoid Them." *Harvard Business Review*, October 24, 2019. https://hbr.org/2019/10/the-risks-of-automl-and-how-to-avoid-them.

must come to terms with what it means to be immersed in technology that is in some fundamental way *not ours*.[2]

To understand this sea change in technology, we need to expand our definition of technical processes to include what would traditionally be called "humanistic" assessments of how information is gathered and represented within machines, as well as how more sophisticated variants of ML algorithms might differ from the broad (and broadly contested) category of "human cognition." The humanities, in short, must develop the language and critical methodologies to read algorithms as texts. This notion is not a new one, and indeed the humanities has already developed well-defined subdisciplines (e.g., "code studies") organized around this very notion of code-as-text. Yet I hope to take a different tack in the work that follows. I deliberately tagged this work with a title—"Reading Algorithms"—that is ambiguous. In the course of this work, I will show that these two words point precisely to where the cross-disciplinary conversation between humanists and technologists becomes vital to answering one of the crucial questions of computational studies: where does the evolving boundary between machine intelligence end and human consciousness begin?

I included the opening dialogue with Siri to suggest why I wish to locate this exploration in the broad concept of reading, and to suggest three ways in which I want to unpack the

[2] It should be noted that some philosophers, notably Heidegger, have persuasively argued that this alienation *qua* causal rupture is not at all new, and indeed in Heidegger's figuration, constitutes an essential trait of all technology. See Heidegger, Martin. *The Question Concerning Technology, and Other Essays*. New York: Harper Perennial, 2013. And other thinkers, such as N. Katherine Hayles, have proposed that all sufficiently complex technical systems—not just ML algorithms—resist human understanding as a system. But Hayles seems to agree with my assessment that an ML system's ability to learn creates an especially provocative relationship human agency to technicity: "Computational media are distinct, however, because they have a *stronger evolutionary potential* than any other technology…" Hayles, Katherine. *Unthought: The Power of the Cognitive Nonconscious*. Chicago: The University of Chicago Press, 2017.p. 32-33.

phrase “Reading Algorithms.” When I asked Siri what reading is, she answered that it is a “complex *cognitive* process.” When asked if she could read, Siri responded that “*intelligent* assistants are *highly literate*.” When asked if she could write, Siri could not parse the question. These three queries correspond to the ways in which I want to explore “Reading Algorithms” and will serve as the organizing basis for the three chapters that follow.

First, I wish to explore “Reading Algorithms” in the sense of what it would mean for an algorithm to read. Algorithms can read in at least one sense of that term—the fact that Siri understood my questions means she understands language both syntactically and as some form of expression of intention. She is, as she claims, literate. But does Siri—an algorithm that we cannot deny “reads” in some form—evidence the “complex cognitive process” that she includes in that definition? This exploration of Reading Algorithms as algorithms-that-read will comprise the first section of this work.

Second, I want to unpack the notion of “Reading Algorithms” to mean a certain formulation of “reading an algorithm” as if it were a text. But am less interested here in reading the code of the algorithm than I am imagining what it would be like to read *as if* I were the algorithm. I maintain that certain algorithms, especially of the ML variety, have reached a point of complexity where it becomes cognitively impossible to understand the *process itself* as a thing that is generative of results. If this is the case, then we have reached the limits of what technical analysis can tell us and must approach these algorithms from a humanistic angle. In his famous essay, the philosopher Thomas Nagel explored the proposition “What is it Like to Be

a Bat?"[3] While Nagel concludes that it is impossible for humans to imagine the phenomenology of a bat, I wish to show that it is productive to imagine what it might be like to read like an algorithm. Exploring this question will open avenues into ways in which humanists can not only understand the inner mechanics of how algorithms generate information about texts, but also give working examples of how humanists can explain how higher-order cognition must be involved in reading like a human as opposed to a machine.[4]

Third and finally, I want to explore what "Reading Algorithms" might mean for humanists to read the *output* of an algorithm as if it were a text. Siri "didn't understand the question" when I asked her if she could write, perhaps indicating what many students of secondary languages discover: it is generally much harder for individuals to produce sentences and texts in non-native languages than it is to read them. And yet there do exist algorithms that output textual data. As we will see, this algorithmic textual output can often mimic its source material with uncanny fidelity. The algorithms that perform this form of mimicry represent some of the most complex instantiations of ML variants in-use today. But sometimes these algorithms fail to mimic their source texts. Literary critics have specialized knowledge about what linguistic or narrative structures differentiate one set of source texts from another, and as such are uniquely positioned to inform technologists *why* their algorithms are failing.

---

[3] Nagel, Thomas. "What Is It Like to Be a Bat?" *The Philosophical Review* 83, no. 4 (1974): 435.

[4] Ian Bogost takes up Nagel's phenomenological questions in his highly entertaining and thought-provoking work *Alien Phenomenology*. See Bogost, Ian. *Alien Phenomenology, or, What Its like to Be a Thing*. Minneapolis: University of Minnesota Press, 2012. Later in "Reading Algorithms" when we turn to the exploration of "reading like an algorithm" I will draw heavily on Bogost's templates. The idea of the phenomenology of things has gained traction in the philosophy of "speculative realism" or "object-oriented ontology," where Graham Harmon and Quentin Meillassoux lay much of the theoretical groundwork for this project. See Harman, Graham. *Object-Oriented Ontology: A New Theory of Everything*. London: Pelican Books, 2018 and Meillassoux, Quentin. *After Finitude: An Essay on the Necessity of Contingency*. London: Bloomsbury Academic, 2017.

This need for a collaborative space represents a very real and immediate concern within the for-profit institutions that control most of the current ML ecosystem.

To conclude this introduction, I wish to situate this work within current academic discourse. Our exploration will bound two fields that have become increasingly central to the academic study of English: Digital Humanities (DH) and new media. Taking the first of these fields, DH as a field seems to be reaching an inflection point, with some academics openly hostile to the inclusion of DH methodologies into literary studies, and others opening much more nuanced inquiries into the validity of the field's tools and results.[5] Since I believe my own studies would be classified as some form of "DH" in the taxonomy of academia, I want to address these objections seriously and continually in the pages that follow. My core observation is that DH has, to this point, been formulated as a mode of statistical discovery—a way of automatically generating datasets from the large corpora of the "great unread."[6] This formula has led to some helpful results, but to conflate DH as a field with statistics is to miss the larger point. And that larger point is precisely the tools—the algorithms—themselves. When we have algorithms like Siri regularly reading *us*, and we increasingly read content that

[5] One of the most insightful commentators on the nuances and potential inaccuracies of DH studies is Nan Z. Da. Da's critiques are particularly illuminating in her knowledge and analysis of the DH "toolkit" from a technical angle. I hope to emulate Da's focus on the tools of DH, but hopefully bringing them to the table as objects of study, as opposed to focusing on their capacity to generate (perhaps specious) statistical results. See Da, Nan Z. "The Digital Humanities Debacle." *The Chronicle of Higher Education*, March 27, 2019. https://www.chronicle.com/article/The-Digital-Humanities-Debacle/245986. Additionally, see Da's piece (and subsequent responses) in *Critical Inquiry*: Da, Nan Z. "The Computational Case against Computational Literary Studies." *Critical Inquiry* 45, no. 3 (2019): 601–39. https://doi.org/10.1086/702594.

[6] The "great unread" is a term originally coined in 2009: Cohen, Margaret. "Narratology in the Archive of Literature." *Representations* 108, no. 1 (2009): 51–75. https://doi.org/10.1525/rep.2009.108.1.51. The "great unread" was popularized by Moretti in the context of DH in Moretti, Franco. "Conjectures on World Literature, NLR 1, January–February 2000." *New Left Review*. https://newleftreview.org/issues/II1/articles/franco-moretti-conjectures-on-world-literature. The term now enjoys wide circulation but, as I shall note later, is sometimes deployed somewhat problematically when using relatively small corpora as a stand-in for "everyday language."

algorithms generate *for us*, it seems as if DH as a discipline should be equally focused on reading these algorithms—in all the [7]senses I mention above—as it is on producing readings based on data collated by algorithms.[8] To this end, I hope to demonstrate an alternate conception of DH that shows how reading algorithms can be used to open new spaces and vocabularies to think about what it means, to use Siri's words, to conceive of reading as a "complex cognitive process."[9]

The second disciplinary intervention I hope to achieve in this work is to explore a concept that is quickly gaining traction in new media—that of the "cognitive nonconscious." The most thorough exploration of this concept to-date has been articulated in N. Katherine Hayles' *Unthought*. Hayles points out that human cognition has traditionally been equated with

[7] The notion of human perception being "written" for us by media operating at timescales below conscious human awareness is the subject of Mark Hansen's work *Feed-Forward*. This notion is further amplified by N. Katherine Hayles' subsequent work *Unthought*, where she suggests that cognitive processes occurring below the "half second" barrier of conscious perception could potentially occupy a much more privileged space within human phenomenology than anything occurring within the slow-time of conscious awareness. See Hansen, Mark B. N. *Feed-Forward: On the Future of Twenty-First-Century Media*. Chicago: University of Chicago Press, 2015, and Hayles, N. Katherine. *Unthought: The Power of the Cognitive Nonconscious*. Chicago: The Univ. of Chicago Press, 2017.

[8] Steven Ramsey, in his influential 2011 work *Reading Machines: Toward an Algorithmic Criticism* pursues a similar goal when he advocates "algorithmic criticism" as a means to use computation to explore the "unfolding of interpretive possibilities" as opposed to a process that is generative of interpretation itself. I wholeheartedly agree with Ramsay here, but I wish to suggest that we can abstract this "algorithmic potential" even one step further: instead of assuming the epistemic stability of something like an "interpretive possibility," we can use algorithms to discover what we actually *mean* when we invoke an "interpretive possibility." In other words, bringing algorithms into contact with literary-theoretical concepts like "close reading," we can approach well-trod debates such as the difference between "everyday" and "interpretive" reading in a fresh and perhaps illuminating manner. Ramsay, Stephen. *Reading Machines: Toward an Algorithmic Criticism*. University of Illinois Press, 2011.

[9] In 2018, Andrew Piper published *Enumerations: Data and Literary Study* that furthers Ramsay's above intervention and brings some of the same issues around DH that I will pursue to the table: "[*Enumerations*] argues that data and computation unquestionable have a role to play in understanding literature, but the way that we have so far approached this problems rests on a number of flawed premises….We still do not have a clear picture of how emerging quantitative methods speak to the questions that matter within the discipline of literary studies." Piper, Andrew. *Enumerations: Data and Literary Study*. Chicago: The University of Chicago Press, 2018. My further intervention in "Reading Algorithms" suggests that the "questions that matter within…literary studies" are also not well defined, and that algorithms can be used to re-frame existing literary-critical problems.

conscious thought. Hayles goes on to demonstrate that much of what we ascribe to "core consciousness" occurs in the space that she calls the cognitive nonconscious. Nonconscious cognition constantly occurs in our minds at timescales below consciousness, such as when we subliminally become aware of a pattern that affects our behavior. Hayles extends this claim to suggest that similar nonconscious cognition occurs constantly in nonhuman and inanimate things. And crucially to our discussion here, one class of cognitive things that Hayles spends a great deal of time analyzing are algorithms.

Unthought not only suggests that the cognitive nonconscious plays a larger role in human thought than many have assumed, but also that nonconscious cognition is much like an algorithm running on a machine. If we take reading, and specifically reading as a manifestation of a uniquely human "cognitive process of decoding symbols to derive meaning," then it becomes essential to delineate what modality of cognition belongs to human consciousness and what belongs to the realm of the cognitive nonconscious. We will try to grasp the contours of this delineation by reading algorithms—not only noting when machines fail to read like humans, but also understanding *how* and *why* machines might cognize texts differently than humans.

To further situate this works' intervention into the academic disciplines of DH and new media, as well as to provide a conceptual background for these three chapters exploring the different formulations of reading algorithms, I propose that we can formulate the delineation between cognitive nonconscious systems and conscious human thought in terms of *memory*. The concept of memory, and especially the externalization of memory, weaves a thread through many of the foundational texts of new media. And the limitations of human memory

lie at the core of DH's claim that we need machines to process vast corpora of texts. To be sure, "memory" is as fraught and complex a term as "cognition." But memory, taken in the context of algorithms, is well understood. And, crucially, understanding how memory functions within reading algorithms will open that essential space where we can think both in humanistic and technical registers to bring forth the *excess* we bring as human readers of texts.

I propose that we componentize the concept of memory—both human and algorithmic—into three components: *sequence, context* and *attention*. The three chapters exploring the facets of reading algorithms will explicate the technical notions of these terms as they manifest themselves in the algorithm. And that will lead us, directly, into how sequence, context and attention differ in the context of human conscious cognition. As I hope we shall see, reading algorithms can not only help us get underneath concepts that have been central to the humanities since its inception, but open ways in which the technical community can decompose the facets of what it might mean to read *unlike* an algorithm.

Chapter 1: Algorithms That Read and the Sequence of Information

Section 1.1: Two Types of Reading

On December 14 of 2016, the *New York Times Magazine* published an article by Gideon Lewis-Kraus entitled "The Great A.I. Awakening."[1] It is a well written and technically accurate piece about the rise of "machine learning," (ML) driven largely by a team of researchers and programmers at Google who collectively form a department called "Google Brain." Lewis-Kraus' article about the rise and rapid spread of the technology created by the team at Google Brain seemed to circulate widely and quickly: within a few days of the article's publication, I received many pings from friends and coworkers that know of my interest in machine learning and language processing. This interest within my small network of acquaintances mirrors a much more widely spread curiosity circulating wherever smartphones and technology are central to everyday life. But it is hard not to notice something riding closely alongside this curiosity. We are curious about what is driving these "AI-like" tools that we interact with on our phones and computers every day, and we want to know what is making many of these apps "smarter" and more functional by small but perceptible degrees. As more applications emerge which emulate and take care of increasing amounts of the more-or-less quotidian functions of our everyday lives, and as these applications become better and better at doing what we formerly managed by hand or by memory, it's hard to repress a corresponding question that

[1] Lewis-Kraus, Gideon. "The Great A.I. Awakening." *The New York Times*. The New York Times, 14 Dec. 2016. Web. 23 July 2017.

threatens to turn into a drumbeat: when will this progression, this improvement, this encroachment of "AI-like" applications—*when will it stop*?

This is becoming a more pressing question as "intelligent" technology driven by machine learning threatens to take away middle-class jobs, just as robotics and software took away unskilled (sic) labor positions in the previous decades. For the most part, the *Times Magazine* article is a balanced review of the technology behind machine learning and the history of this technology, but it contains an Epilogue with a case study of how pattern-recognition software driven by ML algorithms are threatening to replace radiologists:

> Radiologists are extensively trained and extremely well paid, and we think of their skill as one of professional insight — the highest register of thought. In the past year alone, researchers have shown not only that neural networks can find tumors in medical images much earlier than their human counterparts but also that machines can even make such diagnoses from the texts of pathology reports. What radiologists do turns out to be something much closer to predictive pattern-matching than logical analysis. They're not telling you what caused the cancer; they're just telling you it's there.[2]

Given this example, the article moves on to generalize and one can hear the aforementioned drumbeat of anxiety:

> A network built to recognize a cat can be turned around and trained on CT scans — and on infinitely more examples than even the best doctor could ever review. A neural network built to translate could work through millions of pages of documents of legal discovery in the tiniest fraction of the time it would take the most expensively credentialed lawyer. The kinds of jobs taken by automatons will no longer be just repetitive tasks that were once — unfairly, it ought to be emphasized — associated with the supposed lower intelligence of the uneducated classes. We're not only talking about three and a half million truck drivers who may soon lack careers. We're talking about inventory managers, economists, financial advisers, real estate agents.[3]

[2] Ibid., Epilogue, para. 6
[3] Ibid., Epilogue, para. 7

"Inventory managers, economists, financial advisors, real estate agents" and…. English Professors? Why not? When presented with an aesthetic challenge within a work of fiction, or poetry, or film, or whatever the object of interpretation might be—don't we start by trying to recognize *patterns*? We search for literary forms or authorial devices that we've been trained to recognize, search for the repetition of these forms, and think about how adherence to or deviation from these forms can open the way for aesthetic appreciation and interpretation. The article's point is that anything that involves pattern recognition—and it's hard to think of examples of cognitive activity that can't be twisted in some rhetorical manner to be classified as some form of pattern recognition—will be subsumed by some form of algorithm driven by machine learning. In summary, then, after this exceptional article's erudition and accurate depiction of the history of machine learning and its inner workings, the author's final takeaway is a warning: cognitive acts, however complex, can be mimicked by machine learning. What we may think of as some mental process unique to humans can and will be duplicated at some point by an algorithm. *It's just a matter of time*.

Let us suppose that article's most dire warning is indeed accurate, and it's just a matter of time before machine learning algorithms can cover the gamut of natural language processing (NLP) tasks employed by professional interpreters. As it currently stands, certain NLP tasks, such as translation, are performed in many cases sufficiently well to replace humans. Some interpretive tasks, such as the identification of authorial idiosyncrasies, or the identification of features of regional dialect are served well by ML-based algorithms. But on the other hand, similar tasks that might be involved in literary interpretation, such as the detection of an unreliable narrator, or the identification of free indirect speech, are so far away from being

solved that we have no real gauge of whether these are even possible with ML-based algorithms. As long as we have this spectrum of complexity—as long as we know that some problems are *hard* for Machine Learning to grasp—we cannot categorically state, as the *NYTM* article seems to want to do, that it's just a matter of time until all problems are solved by machine learning algorithms.

If all the questions we pose to machine learning are not equal, then, we need to understand what would cause a problem in the domain of natural language processing to be difficult for a machine learning algorithm to grasp. To answer this question, we need to dig a little deeper than the *NYTM* into the actual mechanics of Machine Learning algorithms. The *NYTM* article mentions the seminal "Cat Paper"[4] presented by the Google Brain team that originally drew the world's attention to their research and progress. To summarize, the "Cat Paper" was given at a conference in 2012 and demonstrated the proficiency of Google's Machine Learning algorithms at identifying pictures of cats that were captured from random stills on YouTube. It is worth recapitulating some of the *NYTM* article to give an idea of the difficulty of this problem. As a programmer, if I were given this task of cat-identification, and knew nothing about the techniques of machine learning or neural networks, I would likely start by programming like a good Platonist, thinking of what constituted all the characteristics of an ideal cat and painstakingly encoding them within an algorithm. The article describes this process well:

> You stay up for days preloading the machine with an exhaustive, explicit definition of "cat." You tell it that a cat has four legs and pointy ears and whiskers and a tail, and so

[4] Le, Quoc V. "Building high-level features using large scale unsupervised learning." *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (2013)

> on. All this information is stored in a special place in memory called Cat. Now you show it a picture. First, the machine has to separate out the various distinct elements of the image. Then it has to take these elements and apply the rules stored in its memory. If(legs=4) and if(ears=pointy) and if(whiskers=yes) and if(tail=yes) and if(expression=supercilious), then(cat=yes)[5]

I'd then turn my algorithm loose on random stills garnered from YouTube and immediately be disappointed when presented with an image of an animal with floppy ears, a hairless sphinx breed, or three legs, all of which may represent good, valid, loved cats that don't fit some aspect of my criteria.

Unlike the platonic approach of traditional algorithms described above, neural networks have no ideal criteria, and instead "discover" what it *probably* means to be a cat. This discovery involves many individual software-based "agents" that process an image, each programmed with different criteria to examine a tiny aspect of the overall image. To analogize this process, each agent's processing yields a "vote" representing whether its individual criteria determined if it was a cat or not. At the end of each round of voting, the agents are judged to be right or wrong, and their votes are weighted appropriately. This process is carried out again and again, perhaps millions of times in what is known as neural network "training." Throughout this process, there is a master program that is "judging" each of these agents at the end of each round depending on whether their vote was right. This "judge" then correspondingly raises or lowers the importance of each agent's vote going into the next round. At the end of this training, agents that are always wrong have very little or no say (weight) in the yea-or-nay of catness, and agents that have voted correctly most or even some of the time have

[5] Lewis-Kraus, Gideon. "The Great A.I. Awakening," A Deep Explanation of Deep Learning, para. 4.

correspondingly greater weight. If the programmer can construct millions of agents in a way that each of them are looking at different aspects of the image-at-hand, the laws of probability ensure that the system will eventually reach a state where it correctly identifies images of cats most of the time. The programming of a neural network thus comes down to covering enough *possibilities* of how a cat could appear in a given context as opposed to what a cat *is*. Platonic idealism has been replaced by set theory and metaphysics by statistics; what a thing *is* has been replaced by the construction of unimaginably large permutations of characteristics that will be democratically determined as having manifested, at some point or another, an aspect of an instantiation of a thing.

From a software standpoint, this behavior of neural networks—where they evolve and discover what they are looking for—has always been looked on by some programmers as a species of "magic," and the fact that there are millions of "agents" that need to "learn" or be "trained" before they correctly answer the question has led many to go beyond metaphor and compare neural networks with the activity of real, biological neurons, and to state that the whole process of machine learning via neural networks is the process by which human minds seem to "bootstrap" themselves into cognitive knowledge. But this is a dangerous line of reasoning, where one treats ML algorithms running on a machine in the same manner as a group of neurons clustered in a human brain. This unsubstantiated claim, coupled with the recent quantum-leap in proficiency of ML algorithms, leads to an echo of the *NYTM* article's mantra—it's just a matter of time until every problem solvable by humans can be solved by a neural network with the proper training.

And yet, in the face of this categorical optimism, when one gets down to the actual business of literary interpretation, and the interpreter begins asking some questions that would be commonplace when thinking about some slightly deeper questions about a text, it's difficult to imagine how to even begin to frame the inquiry to a machine learning algorithm. Take, for instance, some questions this interpreter had upon a recent rereading of a particularly beloved novel, Melville's *1851* Moby Dick. The chapters in which Ishmael spends his first few days with Queequeg at the Spouter Inn have always been close to this reader's heart. If one is reading the book critically it is important, I think, to have an opinion on whether Ishmael harbors a homosexual attraction to Queequeg or whether what occurs between the men in their room at the Inn is instead an expression of intimate friendship. This question becomes central when one considers whether the exclusively masculine world of the *Pequod* is a reflection upon facets of what a homo*social* as opposed to a homo*sexual* society might look like. What begins in the comfortable room at the Spouter Inn is the beginning of a larger unfolding that will continue throughout the novel.[6]

This unfolding that will continue over hundreds of pages is condensed into a few chapters, and these few chapters might be condensed even further into a few sentences that, until a recent rereading, had escaped me. In the chapter describing their first night in the Inn,

[6] I am aware that this reading of Ishmael and Queequeg's "honeymoon" is extensively referenced in many other works of literary critical analyses of *Moby Dick*, as is the extended (potential) homoerotism of Chapter 94's "A Squeeze of the Hand." Furthermore, I acknowledge that Melville occupies a privileged space in queer American literary criticism with his subsequent works *Pierre* (1852) and *Billy Budd (*begun 1886). See Creech, James. *Closet Writing/Gay Reading: The Case of Melville's Pierre*. Chicago: University of Chicago Press, 1993 and *The Queer Afterlife of Billy Budd*, in Pérez Hiram. *A Taste for Brown Bodies: Gay Modernity and Cosmopolitan Desire*. New York: New York University Press, 2015 (p. 25-48). My reading here is not intended to break any kind of literary critical ground whatsoever; it is rather a staging of a potential and relatively simple interpretive reading that provides a ready-at-hand example to provide a contrast between "humanistic" and "algorithmic" modalities of reading.

where philosophic Ishmael finally comes to terms with his situation and settles in beside the hyper-masculine Queequeg. The harpooner "roll[s] over to one side as if to say 'I won't touch a leg of ye'" and the night and chapter ends with Ishmael "turned in, and never slept better in my life."[7] They awaken the next morning, with "Queequeg's arm thrown over me in the most loving and affectionate manner. You had almost thought I had been his wife."[8] The next day's action ensues, highlighted by Father Mapple's sermon, and the next evening the philosopher and harpooner once again find themselves in bed together, after having bonded over a few pipes of tobacco, "[Queequeg] pressed his forehead against mine, clasped me round the waist, and said that henceforth we were married; meaning, in his country's phrase, that we were bosom friends; he would gladly die for me, if need should be."[9] After sharing smokes and warmth in their ice cold room, the pair finally get to sleep: "Queequeg embraced me, pressed his forehead against mine, and blowing out the light, we rolled over from each other, this way and that, and very soon were sleeping."[10]

In their first encounter, then, Ishmael "turns in" beside Queequeg, a colloquialism that simply means "goes to sleep," but given all of the emphasis on rolling this way and that, and the fact that Queequeg makes a point of rolling such that he isn't facing Ishmael—and Ishmael "turning in" so that he's facing the harpooner—might this indicate a desire on Ishmael's part for closer contact? And the next night he is rewarded: Queequeg pronounces them "married" (a word that Ishmael could have easily elided from his recounting of the conversation, as it isn't

[7] Melville, Herman. *Moby-Dick; or, The Whale*. New York: Harper & Brothers, 1851. p. 27.
[8] Ibid. p. 27.
[9] Ibid. p. 57.
[10] Ibid. p. 57.

the word Queequeg *actually used* and is immediately followed by the translation into "bosom friends") and they end the night "rolling over from each other, this way and that, and very soon find themselves sleeping." Very soon, but not immediately!

Clearly this two-night stand at the Spouter Inn is heartwarming and contains both fraternal and potentially sexual elements. Whether this relationship is a sexual one is not central to the discussion at hand, but if this were a dissertation about *Moby Dick*, much would depend on whether our reading of this scene, for both Ishmael's subsequent actions toward Queequeg and the dynamics aboard the Pequod might be read very differently if they are construed as coming from lovers as opposed to platonic "bosom friends." Coming squarely back to the point here: how could we possibly expect a machine to track any facet of this discussion, starting with the close reading, but especially into the unfolding of consequences that arise from this close reading. The complexity of this exercise seems as far away from the cat recognition as summing a column of numbers in a spreadsheet differs from performing the calculations necessary to put astronauts onto the moon. But computers have indeed solved both problems, and this analogy gets at a point that we must consider. Is the gap between this type of interpretive-based close reading and the cat-recognition or other Machine Learning feats that take our breath away—is this a gap merely of *complexity*? In other words, is my reading of this scene between Ishmael and Queequeg, and the suggestion that it would unfold into something more fundamental and central to a holistic interpretation, simply more *complex* than recognizing an image of a cat? Or are we dealing with problems that are incommensurate; of not only different orders of complexity, but different problems entirely? This, I think, is precisely the question we need to press on and can only be answered by a deeper investigation

into how Machine Learning algorithms formulate problems, how they process and consume their input data, and how, to put it in human terms, machines "want" their questions to be put to them.

Consider the ML-based cat-recognizer again. What, exactly is this algorithm using as its inputs and how is it manipulating those inputs to arrive at its answer? We described earlier the millions of software-based agents that comprise the neural network that serves as the machine learning algorithm's infrastructure, and how these agents are each allowed a vote at the end of each round. This makes sense and serves as an excellent metaphor to explain how a neural network is trained at a high-level. But we need to go one step lower: these agents are pieces of computer code, so what do they "look" at? What are their inputs, how do they differ from one another, what do they output, and how are these outputs organized into an answer that collectively seems to exceed the complexity of what each individual agent is doing?

First, the agents need to be operating on the same piece of data. In this case, the agents are all looking at the picture of a cat. But what is a "picture" to these agents, and what does "looking" mean? The picture to these agents is a matrix of pixels, a square of, say 256 by 256. Each pixel has a position within the matrix—say in the first row, and 50th column—and numbers representing hue, saturation, and value (color, darkness, intensity). These values—which we can just refer to for simplicity's sake as "hue"—in software can be condensed into a single number, with, say, 1 being black and 65,536 being white.[11] From the agent's perspective,

[11] Colors in code are in practice usually expressed in hexadecimal format, with #ffffff being pure white and #000000 being pure black, but a simple integer seems more relatable to a nontechnical audience than a hexadecimal expression.

then, each point within the image has two informational dimensions: its position within the matrix, and its hue. Now suppose we want an agent that looks for a circle in the picture, perhaps in the thought that this might likely represent a cat's face. We can do this by having the agent look for a certain configuration of dark pixels in the matrix that would indicate a circle. But, of course, sometimes the cat's face in the picture might be turned to the side, so it's more of an oval. That's no problem, we can create another agent that looks for this trait. Sometimes agent #1 will be right, and sometimes agent #2 will be right. Because we can create millions of agents, each looking for different configurations of hue and position, we don't have to worry about the correctness of an individual agent's criteria. We just need to make sure we create *enough* agents—cover enough *possible* features of what the cat-picture will look like. Because we have millions of these agents, and can have them vote millions of times, it doesn't matter how likely it is that an *individual* agent is constructed to recognize "catness" correctly; it simply matters that we cover enough potential variations of a cat-image so we encompass the many thousands of ways different cats can appear in different images. If an agent is always wrong—say it has instructions to look for pixels that would indicate a face as a perfect square—that's not an issue: that particular agent's vote will just get weighted further and further down in the training process so that its opinion eventually becomes worthless.

Here is the key observation: this entire machine learning process to recognize cats works *only because* we can flatten the thing we are looking at into a very small informational subset: position and hue. The ability to flatten the problem at hand – this ability to reduce what the agents are looking at – is a necessary precondition of all machine learning algorithms. If we can flatten the data in such a manner, then at a certain level of abstraction all our inputs

appear as homogenous. As a result, all our software-based agents can operate in a homogenous manner. In machine learning systems, we operate in a world where questions are very rigorously limited, answers can only be considered on the basis of a very limited subset of traits, and the only possible outcomes are "yes" or "no." From this standpoint, it is amazing the breadth of problems that can be addressed with ML algorithms given the amount of reduction that is occurring to the input data. When a music application attempts to learn your preferences, a song is decomposed into notes: data with amplitude, pitch, duration and positionality within the song. When another algorithm attempts to identify an author, the text is broken down into words and those word's positions within the text. In all of these cases, we are performing massive reductions on the entity the algorithm is looking at, yet it can recompose very sophisticated answers as if it were considering a much larger subset of information. Given this, it is no wonder people are tempted to attribute a certain level of human-like intelligence to these algorithms.

And yet, when pressed, the limitations imposed by this informational reduction are bought strikingly into the forefront. In each case, what seems to be lacking in the algorithm is some notion of unfolding – some notion, to be more precise, that the thing you are asking about occupies a *nonlocal duration* and is comprised of *nonhomogeneous input*. An interpretation is not simply a collection of static data that have completely symmetrical cross-correlation, like the pixels fed into an image recognition algorithm or a collection of digitized notes fed into a music preference analyzer. And from a literary-interpretive standpoint, nowhere is this limitation more striking than when asking aesthetic or interpretive questions about a text. Consider the question of the nature of Ishmael's and Queequeg's relationship:

there is a definite sequence that occurs, and furthermore occurs from the point of view of Ishmael. Ismael seems to "warm" to Queequeg more quickly, perhaps seeking warmth first as he "turns in" on their first night together. Queequeg eventually reciprocates, but he certainly does not do the same for every sailor he shares a bunk with—the harpooner was, after all, alone when the two first met and ends up splitting half of his worldly earnings with Ishmael, a thing he can probably only do once. Therefore, something must have occurred within Queequeg as well – some recognition of a special quality in our narrator, a process that occurred between the first and second nights. But, again, note here how we not only have a question framed in time—an question of how events are sequenced and how readers arrived at meaning from this sequencing—a question that requires us to think in different registers of time. How this unfolds in readers as an aesthetic moment requires a consideration of how these events were narrated to us by Ishmael, a guess at how they may have unfolded for Queequeg given what we know about him as a character, and some understanding of how warmth and affection take hold and unfold in real affective life.

In the middle of these nights, Ishmael reflects on sleeping in a cold room, and he's of course talking to us about more than physical warmth:

> because truly to enjoy bodily warmth, some small part of you must be cold, for there is no quality in this world that is not what it is merely by contrast. Nothing exists in itself…For the height of this sort of deliciousness is to have nothing but the blanket between you and your snugness and the cold of the outer air. Then there you lie like the one warm spark in the heart of an arctic crystal.[12]

[12] Melville. p. 53

And here, perhaps more than anything else, is the challenge laid out before us when we speak of machines truly understanding an aesthetic moment. We have unfolding temporality in multiple registers, with the body in between, and without understanding this, as well as how the above text, sitting right in the middle of Ishmael's and Queequeg's "honeymoon," can relate all of this together, the project seems altogether insurmountable. Is there any way for a machine to recognize the way in which time comes into play in aesthetic experience, or does the informational reduction within ML algorithms always imply a form of stasis? It turns out Ishmael might have a hint for us: "there is no quality in this world that is not what it is merely by contrast. Nothing exists in itself."

Section 1.2: Pattern and Sequence

In their influential 2016 article "Literary Pattern Recognition: Modernism between Close Reading and Machine Learning"[13] Richard Jean So and Hoyt Long apply ML techniques to classify texts as Haiku within a given corpora. Haiku as a form is defined by a certain sequential arrangement of words and syllables, a pattern that So and Long show can be identified by ML algorithms. The Haiku pattern also manifests itself in many cases as an aesthetic quality. A casual reader of haiku often experiences a reflexive feeling of profundity in this sequencing, a feeling perhaps so universal that it has given rise to a parallel genre of *senryu*, which twists the form for humorous or satirical purposes. In their DH work, So and Long have therefore created a ML algorithm that identifies a form of sequencing that is essentially temporal in nature and is also connected to an aesthetic feeling. While the algorithm is obviously unaware of the

[13] Long, Hoyt, and So, Richard Jean. "Literary Pattern Recognition: Modernism between Close Reading and Machine Learning." Critical Inquiry, vol. 42, no. 2, 2016, pp. 235–267

aesthetic nature of each poem, if sequencing here is directly correlated to haiku (as it manifestly is shown to be) and if this sequencing is normally evocative of a certain emotive contentment, could it simply be the case that the narrative sequencing I outline above in *Moby Dick* is simply more *complex* than the sequencing within Haiku? And if that is the case, could it simply be that we just need ML algorithms to understand more about *time* to decompose more complex narrative structures into informational units that can be analyzed by the machine?

As we turn to a more nuanced discussion of how ML algorithms can model time, Ishmael's quote at the conclusion of the last section comes back to us: "there is no quality in this world that is not what it is merely by contrast. Nothing exists in itself." Speaking from underneath the warmth of his blankets and close companionship with Queequeg, Ishmael is aware only of his comfort as a contrast to the larger winter chill of his room. But after his harried and anxious initial meeting with Queequeg, followed by this slow-paced luxuriating in bed as he shares a pipe with the harpooner, Ishmael might as well be talking about time itself. To follow Ishmael here, does a humanistic assessment of time function in different registers; are we, in other words, only aware of time through the existence of some cognate to our direct immersion in an extrinsic, ambient condition? Paul Ricoeur, in his eminent three-volume work *Time and Narrative* takes up this very question.[14] Ricoeur, whose analysis I cannot begin to do justice to in the space we have here, draws upon Aristotle, Augustine, Kant, Husserl and Heidegger to explore the proposition—addressed in one way or another by all of the aforementioned thinkers—that there exist at least two modalities of time: that of

[14] Ricoeur, Paul. *Time and Narrative*. Chicago, IL: University of Chicago Press, 1984. This analysis mostly draws from the third volume in this series, which contains a useful summary of the first two volumes as well as the descriptions of the mimetic structures of narrative time that I will draw upon here.

"cosmological time"—the universal time that surrounds us like the cold in Ishmael's room—and "phenomenological time"—the time we experience as humans in the world.

Ricoeur's intervention into the longstanding questions around the relationship between cosmological time and phenomenological time is to tie the latter to the structure of narrative itself. Bluntly stated, one facet of Ricoeur's position is that interpretive reading mirrors the sequencing intrinsic to phenomenological time in two ways. First, in the sense that we order events in our memory to read ourselves as if past events were contributing to a continually ongoing narrative of our present. Second, as the much stronger position that some internal conception of narrative itself must exist *a priori* for us to experience any form of internalized time. A corollary to the first proposition is that narrative time runs in two directions: we encounter events sequentially and this is always a forward movement through our "story," but we must also retroactively order past events to understand our present as a story, as the arc or coherence of narrative only emerge as the result of retrospective interpretation. But the corollary to the second position contains an aporia: if internalized time is the result of an ordering of events into this bidirectional structure of narrative, but narrative *as a cognitive facility* arises from our ability to sequence time in this bidirectional, phenomenological manner, it seems impossible to bootstrap any concept of time as narrative without first knowing how to arrange time in the service of becoming a narrative.

Heidegger proposed the ontological conditions of being-toward-death and care to get underneath this aporia. Ricoeur builds on this by describing a series of mimetic structures inherent within Being that give rise to a form of intuitive sense of narrative. These are extremely involved and difficult propositions. What I want to show here is how our

understanding of how information is represented in ML algorithms—their own version of sequencing and information reduction that we explored in the previous section—can allow us an alternate, and perhaps more tractable, way to reach Heidegger and Ricoeur's solution to this aporia. But beyond simply offering an alternate method of understanding of these problems, I want to show that we can reduce these complicated notions of cosmological time and phenomenological time into a difference of how we conceive of the concept of sequence. In this alternate formulation, I hope to come squarely back to our question of what machines are missing in higher-order reading.

Put into context of So and Long's ML analysis of haiku and the relative interchangeability of time and sequence in their classification algorithm, one is tempted to say that the complex notion of "different" times espoused by Ricoeur, as well as our brief foray into the intricacy of narrative time as it exists in *Moby Dick* is precisely where algorithms fail. But that answer glosses over the fact that algorithms *can* account for time in different registers. In the introduction, I stated that algorithms can provide humanists a way of getting underneath difficult concepts, such as those that Ricoeur proposes. To this end, allow me to explain a real-world algorithm that does indeed account for different modalities of time, which will in turn provide us with an opening to understand what might be so intractable about capturing phenomenological time.

One of the earliest adopters of ML technology was by traders wishing to identify patterns in the electronic marketplace. These analysts gather oceans of data—every price fluctuation in the oil markets, or a group of stocks—over years. In its simplest incarnation, this analysis attempts to identify patterns in the marketplace that would precede substantial up-or-

down movements in a certain stock or commodity. The informational units that these researchers decompose their problem into are varied—they could include the magnitude of the price unit, the volume traded, or correlated movements in other stocks or commodities. This analysis is inherently sequential: it takes the ordering of past units of information as a predictor of what is to come next.

Now if one treats time in this ML-based market analysis as a homogenous quantity—as the "cosmological" time of Heidegger and Ricoeur—one will immediately find a pattern of periodic, extreme price fluctuations at precisely defined intervals. This is due to a very simple real-world fact: companies regularly release earnings reports on given dates which result in large movements in the stock; commodities markets swing wildly on scheduled agricultural or energy reports. Now one may not know if the price will fluctuate up or down, but there are methods to profit simply from the knowledge that there will be a period of fluctuation. There are financial instruments—options—that are directly tied to the rate of fluctuation, or volatility, of the underlying asset. Market participants can purchase the option to buy 100 shares of IBM stock at $150 at some point within the next year. How much should this option cost? Obviously one factor is the price of IBM stock right now (currently about $145). But more deeply, the price of this option should be tied to the volatility of the asset. If the price of IBM fluctuated wildly from day-to-day, as something like Bitcoin, this option should cost more, as there would be more uncertainty that IBM could indeed rise above $150, allowing the owner of the option to realize a profit.

An ML algorithm that treated time as a homogenous quantity—cosmological time—in this scenario would always discover a pattern in which the price of an option would rise before

a known event. But this information is not intrinsically valuable because every informed market participant knows when the event will occur and will raise their prices accordingly. The goal of the ML trading algorithm is to detect complex patterns that are unexpected, so it must be modified in some way to ignore these known events. This leads us back, albeit along an odd route, to Ricoeur. If we take these patterns of scheduled market events as some form of extremely simple narrative structure—like a children's book that keeps returning to the same resolution (Dr. Seuss' "I do not like them, Sam-I-Am")—then we have some claim to a form of phenomenological time that arises from simply knowing the story. And, indeed, one way to fix this ML algorithm so it does not mistakenly identify this pattern as a potential opportunity is to extend its accounting of time during these events. In other words, one could simply adjust the ML algorithm's input to treat the 5 minutes around a market event as if that event lasted several hours. This, in effect, would tell the algorithm to expect as much movement (volatility) in these five minutes as would normally occur within a much longer timespan of regular market activity.

One could conceivably program a variant of this phenomenological time into an ML algorithm that analyzes textual data. Consider Ishmael and Queequeg again. In Ishmael's narrated text, we can identify many words in his initial foray into Queequeg's room that indicate anxiety, fear, dread:

> The *devil* fetch that harpooner, thought I, but *stop*, couldn't I *steal* a *march* on him—*bolt* his door inside, and *jump* into his bed, not to be *wakened* by the most *violent knockings*? It seemed no *bad* idea; but upon second thoughts I dismissed it. For who

> could tell but what the next morning, so soon as I *popped* out of the room, the harpooner might be standing in the entry, all ready to *knock* me down**!**[15]

And then, in the previously quoted denouement of the episode:

> because truly to *enjoy* bodily warmth…For the height of this sort of *deliciousness* is to have nothing but the blanket between you and your *snugness*...[16]

There are natural language processing tools to perform "sentiment analysis" based on certain sequences of emotionally charged words, so it is possible to assign some score of "urgency" or "anxiety" to these passages based on the highlighted keywords. And furthermore, ML pattern matchers do not strictly care about the directionality of the sequence, so in the same manner that Ricoeur notes that narrative phenomenological time retroactively conveys meaning to past events, the ML algorithm could identify that the warm sentiment at the end of the episode was correlated to and modifies the anxiety in the passage that came before.[17]

With this thought experiment we can see that we could, potentially, modify ML algorithms to identify some facets of phenomenological time as it manifests itself in a narrative. And we could, conceivably, annotate moments within narrative time that potentially

---

[15] Melville. p. 22 (emphasis mine).
[16] Melville. p. 53 (emphasis mine).
[17] Sentiment analysis was widely used in the trading industry as early as the mid 2000's to quickly and automatically parse and generate trading signals from RSS-based financial news feeds. These tools are commonplace now, although it can be extremely difficult to configure these tools to generate consistently reliable signals in the financial markets. Sentiment analysis tools are now common in DH analyses as well. Antonio Moreno-Ortiz, a DH researcher from the University of Malaga, has gone so far as create a UI-based sentiment analysis tool for use by the general public: Moreno-Ortiz, Antonio. "Lingmotifs: Sentiment Analysis for the Digital Humanities." *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, 2017. https://doi.org/10.18653/v1/e17-3019. Perhaps the most widely used toolkit used in sentiment analysis is the SentimentAnnotator package that comprises part of the wide-ranging Stanford CoreNLP suite of tools: https://stanfordnlp.github.io/CoreNLP/sentiment.html.
.

correspond to affective readerly moments. This can all be extrapolated from some form of algorithmic identification of sequencing. Insofar as we can conflate sequence with time and narrative, and furthermore insofar as we can reduce interpretive reading with the identification of narrative and its relationship to interior time, it seems like we have set the stage to argue that ML algorithms can indeed approach higher-order cognitive engagement with texts. Yet, we seem to have a case here where the whole is far greater than the sum of its parts. *So what* if we can algorithmically identify rising and falling moments within a narrative? *So what* if we can potentially program a ML tool to identify that there could be a platonic or sexual relationship between Ishmael and Queequeg? What would an algorithm be able to do with this information? Coming back to our introduction, I can imagine at some point asking an enhanced Siri what happened between Ishmael and Queequeg and receiving the response "Ishmael was initially afraid of Queequeg, but they eventually became close friends." I cannot ever imagine receiving an answer to the question "Does Ishmael and Queequeg's relationship make Ahab's quest more or less insane?"

One could certainly argue that the latter question is simply more complex or ambiguous. But I wish to claim that there is a *qualitative* difference here, and this difference is more than a quantitative increase in complexity. And furthermore, the very way I would need to sequence information to *think* about this question is entirely different than how ML algorithms need to sequence data to arrive at information. The key difference at this stage of the analysis can be stated as follows: in ML algorithms, *sequences can only arise as orderings of informational units* and the *informational unit must emerge before the sequence*. In our original example of the seminal Google cat-recognizer, we decomposed the data into pixels and sequenced the data

because we wanted to look for a cat. In the example of the options trading algorithm, a more complex sequencing of the variables of time and volatility was entangled *in order to* identify instances where options were genuinely over- or under-valued. In our thought experiment where we would identify different affective narrative states within *Moby Dick*, we chose sentiment as our primary marker *because* we knew Melville uses these to achieve a certain affective response. In all of these, the unit of information—what constitutes the sequence and pattern itself—must be configured ahead of time, as a pattern cannot arise until a homogenous component is determined to constitute the sequence.

What is missing in the claim that ML algorithms can model a human assessment of phenomenological time is precisely this malleability in the units that comprise an experience of sequence. To be sure, at some level, humans must sequence information in the process of reading *a priori*, as we could not parse sentences and grammar if we approached a text with no notion of it as a sequential object. Yet in the emergence of narrative in higher-order reading, the reader must not only order what has come before in the service of what they are experiencing now, but also choose *what* comprises that ordering. That ordering could simply be temporal, a reconstituting of events in the text in a sequence that explains the current moment. But I could just as well order the narrative in terms of affective moments, or as shifts in narrative tone or foregrounding of characters. To take our literary-critical example above, I believe a thorough examination of the relationship between Ishmael and Queequeg and its relevance to Ahab's monomania must take into account how Ishmael's genial first-person narration fades in some chapters, when those chapters occur, and who or what is foregrounded in those chapters. This sequencing is essentially combinatoric, meaning that a pattern

constituted *solely* of these units of information makes sense only when placed alongside other units that would constitute different sequences.

Section 1.3: Picturing the Cognitive Difference Between Pattern and Sequence

To be sure, the above formulation of how informational units are composed in the process of interpretive reading is a complex one. To ground this concept and to help us explore how informational units are tied to qualitative differences in cognition, let us consider another concrete example. In the spirit of Google's image-based ML cat-recognizer, we will leave the realm of reading for a moment and take two images. Consider an image of a tapestry that one might associate with a "Persian" rug:

Figure 1. Example of a "Persian" rug for sale on Amazon.com

Contrast this with Seurat's famous pointillist work that hangs prominently in Chicago's Art Institute, "*A Sunday Afternoon on the Island of La Grande Jatte*" (1884)[18] :

[18] Seurat, Georges. *A Sunday on La Grande Jatte.* 1884. Oil on canvas. 81 in. x 121 in. Art Institute of Chicago, Chicago. https://www.artic.edu/artworks/27992/a-sunday-on-la-grande-jatte-1884

FIGURE 2. SEURAT'S PAINTING, LA GRANDE JATTE

Imagine attempting to program an ML algorithm to determine some sort of pattern in the above images. Crucially, I must first specify what I believe constitutes the informational unit to the ML program up front as the algorithm needs to sequence the data before it can arrive at a pattern. But given what I, as the programmer, know about the ontological status of "Persian" rugs and pointillist paintings, this is straightforward: in the case of the rug, I want the code to consider each thread's position and color; in the painting, I want the code to consider the position and color of each of Seurat's "points." Since I have collapsed both entities into identical informational units—position and color—I can now, from a programmatic standpoint, ignore the fact that these images represent two entirely different things in the physical world.

In this way, the algorithm elides the underlying media in a transformation into pure information.

This elision of things into pure information would conceivably allow an ML algorithm to see a pattern in both the rug and the painting. But qualitatively different patterns emerge within these analyses. In the case of the rug, the pattern that emerges is a *tessellation*, which is a repeated, geometric sequencing of color and position, manifesting itself in the precise positionality of colored threads. By virtue of their repetition and geometric stability, tessellations—or groupings of tessellations such as those expressed in the different regions of the rug—can be expressed mathematically. Because of this repetition and reducibility into a deterministic mathematical expression, an ML algorithm could potentially grasp the totality of the *informational* content within the rug-as-informational-pattern. To put it another way, given enough training, an ML algorithm could learn the quiddity of the pattern of the rug, the essential traits that would allow the code to cognize and reliably determine *this* image represents the example rug, whereas *that* image does not.

Now in Seurat's painting there are also local, deterministic patterns that emerge by virtue of the painter's reliance on color theory to compose the overall image. In certain regions of the painting, such as the foreground of shaded grass, Seurat relies on precisely positioned squares of paint that, given enough distance between the eye and the painting, coalesce into a dark green expanse. Given a local area of the painting, then, an ML algorithm could be trained to correctly infer the probable color *given* its position within the painting. But wholly unlike the tessellation of the rug, the transitional portions of the painting, such as when the dark grass gives way to the reclining man with the pipe in the leftmost foreground, are *meaningful* in a

way that wholly transcends the transitional states in the rug. In the rug, the thin, single-hued dark edge gives way to a similarly thin, geometric spacing of squares, each of which expresses a repeating tessellation of threads. There is no "higher order" meaning to this transition; it is simply a facet of the overall pattern of the rug itself. But in Seurat's painting, the transition of the dark grass into the pinkness of the man's arm contributes to *something else* that is not a pattern, a feature that ultimately reveals and necessitates a qualitatively different cognitive engagement with a sequence of color and position.

We must be extremely careful here, as we have arrived at a very reduced space where we can formulate what I have termed "higher order" meaning above—a term that I deploy to differentiate the nonconscious, mechanistic identification of sequence-as-pattern within nonconscious ML algorithms from the conscious, human experience of sequence-as-aesthetic. Expressed bluntly, the key difference between the rug and the painting is that we cannot assign any form of cognitive *expectation* to the transitions within the different areas of the rug. The solid border yields to the tessellated pattern of squares *because of* nothing. On the other hand, in the painting we can understand the shift in sequence between the points comprising the grass and the assemblage of sequential colors that define the man *because* it "makes sense" that a man might recline on the grass on a sunny day beside the Seine. "Makes sense" here is used as a shorthand for the cognitive facility of assigning a probabilistic notion not only to the experiential reality that one might expect to see a person in such a position on such a day in such a locale, but also the expectation that such an entity belongs to the aesthetic of the painting as a whole. This aesthetic manifests itself as another, but quite different, form of sequence, the notion that, far above the sequential placement of squares of paint to arrive at a

noncognitive perception of color, there exists a *nonuniform* sequence of things within the painting—the expanses of shaded and sunlight grass, the blue of the Seine, the people and animals basking in their own idiosyncratic pleasures of the day—that arise *as a sequence* only as a retroactive consequence of contemplating the painting as an experiential whole.

While the notion of aesthetic expectation as a higher order form of sequence gets us near to the heart of what we are after here, a key objection to how I have formulated this difference above will complete the arc of this chapter. One can certainly argue that there exists a similar juxtaposition of nonuniform sequences in the rug, and that this juxtaposition of patterns, however meaningless in and of themselves, "makes sense" as part of a wholly different aesthetic belonging to "Persian" tapestries. And one could press this point further by stating that there exist whole genres of art—surrealism, abstract expressionism, et. al.—in which transitions make no sense in terms of experiential probability within the world. Both objections are correct. There is the temptation, then, to conclude that what I have termed "higher order" sequencing is synonymous with aesthetics itself—i.e., that the only way in which sensibility emerges as a higher order property of an assemblage of sequences is through the association of an essentially human, affective response to those sequences. If, as Henry James has written, "in the arts, feeling is always meaning," can we follow through with the saccharine conclusion that the higher order meaning humans assign to art and, from the opposite direction, the things I have shown that ML algorithms miss within art, simply exist by dint of the (in)ability of machines to emulate or cognize human feelings?

This assessment may be true, but it gestures towards a metaphysics of feeling that I wish to resist for the moment, as we are now positioned to ask *why* the cognition of aesthetic

or affective content might be complex. Consider how, in the last sections, ML algorithms cognize their informational space as probabilistic outcome determined by a pattern. A pattern, by definition, only emerges as a repetition of information, and is subordinated *a priori* to how that information is organized *qua* a cognizable trait. In our example of the rug and the painting we have seen, in the position and color of thread and paint, there exists a level of lower order sequential information that machines could potentially cognize to some end. At this low level, sequences can be subordinated to patterns; they yield information only in their repetition. But at the higher-level, sequences exist as top-down organizational structures—structures, in other words, are not placed in relation to one another deterministically to affect the repetition necessary for a pattern. Instead, we, in a cognitive and ultimately human capacity, sequence higher order structures such that they emerge *as a sensible result* of an affective or aesthetic organization of the thing taken in its entirety. The man with the pipe in the foreground of Seurat's painting is there because it is a sunny day and the Seine is near; the Seine is not there as a result of the man and his pipe. Similarly, but more abstractly, I cognize the dark thread at the edge of the rug as an aesthetically pleasing way to contain and anchor the complicated tessellations within; it is there in its simplicity *because* of the complexity of the interior.

In this way, aesthetic appreciation subordinates sequence to a larger, holistic appreciation of an object. Yet it also paradoxically and retroactively justifies the sequence as contributing to that same appreciation of the thing as a whole. With this, we come squarely back to Ricoeur's aporia in time and narrative discussed in the prior section: some notion of phenomenological time is necessary to understand events as unfolding within a narrative, but our capacity to understand narrative as a structure is predicated on some understanding of

phenomenological time. What we have done by approaching this question from the technical perspective of ML is show, concretely, how this aporia can be decomposed into qualitative gradations in sequence. There are low-level sequences—patterns—that are cognized, perhaps nonconsciously, which form the basis of things occurring sequentially *because* of a deterministic organizational structure. And we can thus picture higher-order sequences that result from—and are simultaneously constitutive of—aesthetic or affective appreciation as reifications of a desire to find a similar causal structure within our experience of a thing as a coherent whole.

Section 1.4: Conclusion

In this discussion of sequence and how sequential information is processed by ML algorithms, I have explored the gap between the "reading" that algorithms currently perform and what we, as humanities scholars, would consider full-fledged interpretive reading. In this analysis, I hope to bring what I have loosely termed "higher order" interpretation into some alignment with the difficult problems that Ricoeur addresses in *Time and Narrative*, which is to picture interpretation as a retroactive organization of information based on an emergent understanding of an overall structure. By approaching this investigation by first exploring how ML algorithms cognize information, as opposed to beginning with a humanistic theory of phenomenology, I hope to have shown how *time*—which is intensely problematic for Ricoeur and other philosophers of hermeneutics—can plausibly be reduced and discussed as an artifact of sequence. ML and other nonconscious systems cognize time as an ordering, as some form of sequence that is usually imposed by the programmer *a priori.* Thinking of time in these mechanistic terms is helpful to dissociate the metaphysical or ontological "givenness" of time

from what it ultimately must become in the process of algorithmic interpretation, which is simply another dimension of information. By thinking of time in terms of sequence, and by proposing two modalities of sequence—the lower order subordinated to pattern, the higher order subordinated to an aesthetic whole—I propose that it is the latter that proves to be resistant to algorithmic reading. I have followed Ricoeur to state that higher order sequencing is fundamentally retroactive and processually emergent. But I hope to have shown that it is not the *a posteriori* structure itself that presents difficulties to algorithms, but rather the fact that this higher order sequence is not a pattern—that, more precisely, a higher order sequence cannot be understood as the causal result of any structure wholly contained within the thing itself. Thinking of why a machine cannot cognize a sequence that is not a pattern *as a unit of information* is a productive way of stating this problem across both the technical and humanistic community.

To conclude our discussion, I want to come back to reading, and specifically the question I asked of *Moby Dick* at the end of section 2: Does Ishmael and Queequeg's relationship make Ahab's quest more or less monomaniacal? I think the answer is absolutely *more*: however one reads the warmth in the scenes between Ishmael and Queequeg at the Spouter Inn, this relationship contains an essential humanity, a grounding that comes into stark contrast to Ahab's initially metaphysical and, eventually, horrifically literal bond to the whale that comes to symbolize everything alien and irrational within a mind gripped by obsession. But this interpretation represents knowledge that only fully resolves itself at the end of the novel as Ahab is dragged down into the depths, and afterward, as we try to synthesize character, plot and dozens of other formal artefacts of the novel. One pieces together these

many threads, all comprising vastly different types of informational content, and one could find a reason, or confer some form of sequence or cause to these heterogeneous events, as I have done in my short critical statement above. I could now, conceivably, program ML tools or other algorithmic tools to identify many of these various traits. But these exercises, however interesting they might be, are essentially configuring and writing code to "discover" textual features that I know are already there, or that I think would contribute to a structure that aligns with my interpretive assessment. The gulf between identifying patterns and discovering a holistic sequence of events that organize a reading into an interpretation is like comparing Ahab's quest to Ishmael's. Ahab knows—and perhaps has always known—what he wants before he steps foot on the *Pequod*. Ishmael begins distraught and aimless, bears witness to events wholly irrational and beyond comprehension, and is left, at the end of the novel, wondering "Why then here does any does any one step forth?" But we know that Ishmael will take that step, and his journey is the essence of humanistic interpretation writ large.

Chapter 2: Reading Like an Algorithm and The Role of Context

Section 2.1: The Links Between Aesthetic, Context and Merit

In the previous chapter, we examined the role of sequence in both algorithmic and humanistic analyses of texts. We concluded that there are two qualitatively different manifestations of sequence: 1) a local sequencing contained within the object that manifests itself as patterns of information and 2) a more complex, higher-order sequencing that emerges as a top-down organizational structure when the object is considered as a whole. I suggested that this higher-order sequencing is where algorithms-that-read fail, and as such delimits a concrete location where human cognition is doing *something more* than machines running code.

In the above formulation, when I invoke the "object considered as a whole" and a "top-down" consideration of the information contained within a text, there is the hint that we need something "larger" than the text itself—an extrinsic consideration of the object as it is placed in relation to other objects within the world. Another way of stating this might be simply that we need some notion of *context* to perform this higher-order sequencing. In Seurat's *La Grande Jatte*, that context might involve an embodied knowledge of what it might be like to enjoy that particularly enjoyable Sunday. It is pleasing to see the transition from the dark shaded grass in the foreground to the lighter, sun-drenched grass beyond, as it recalls the sensual pleasure of having the choice to bask in the sun or to cool down in the shade. One can imagine the calming susurration of the Seine as it flows past. Alongside this vast set of references arising from bodily experience, there are many other forms of context in play here as well. The mixture of people from different social classes within the park—the sleeveless man with the

pipe in the foreground lounging in close proximity alongside a couple in formal dress—further reinforces the tranquility of the image: tensions associated with class conflict might be temporarily set aside in the mutual enjoyment of the day. This extrinsic contextual information also organizes a feeling of deep appreciation for the intrinsic formal qualities of the painting itself. Even in today's age of CGI and other advanced imaging technologies, I continue to see the same look of delight on the faces of visitors to the Art Institute of Chicago who, moving away from the painting, witness the transition from individual pointillist squares of color into the sweeping scene of the park.

I dwell on the painting to demonstrate my own appreciation of its qualities as an aesthetic whole, and to illustrate how that aesthetic organizes an appreciation for the local manifestations of sequence within the painting—the placement of the ostensibly working-class pipe smoker alongside the well-dressed couple next to him, the awesome skill and prolonged concentration it must have taken Seurat to harness pointillism to create this scene, etc. But more importantly for the purposes of this chapter, I wish to illustrate how deeply *context* is embedded within my assessment of the painting's organizing aesthetic. I deployed many sources of contextual information, some of which emanated from my status as an embodied human, others that arose from my limited understanding of *fin de siècle* French society. So if it is true, as I have claimed in the last chapter, that cognition of higher-order sequence is tantamount to understanding an organizing, holistic aesthetic, and furthermore if it is the case, as I have suggested in the above paragraphs, that this aesthetic is rooted to great measure within extrinsic, *contextual* information, then it is fitting that our next task should be to figure how—and to what extent—algorithms can cognize context.

Unfortunately, as was the case with the phenomenological experience of time in our first chapter, "context" is a fraught concept that has traversed a complicated path in many disciplines, and nowhere more so than in linguistics and literary studies. In the latter case, I will simply point to the generation-spanning debates within the profession regarding context in relation to interpretive activity. The historical biography of the author, the author's imagined intention, the larger historical milieu under which the text was produced, the formal qualities of the text itself in relationship to the authorial norms of the day, the reader's individual orientation toward the text—all of these, and many more, have been proposed as relevant (and in some cases, the *only* relevant) context to consider in interpretive activity. Because the word "context" carries so much baggage, and since I wish to investigate context as an *informational* quantity and the (in)ability for context to be fully cognized by machines, I propose to transpose this analysis of context as an abstract quantity onto the more concrete question of literary merit.

While literary merit is not always synonymous with a text's ability to reproduce the historical and humanistic contexts that encompass its production, I believe it is fair to say that many canonically meritorious texts become canonical in their ability to deftly weave social, historical, and personal contexts in order to address larger concerns about the human experience of being in the world. To take a single example, Henry James is regarded as the quintessential novelist of manners, but to understand why he is assessed as such requires a deep understanding of the contextual protocols of the society and age that inform his writing. Through James' deep understanding and subtle twists of these historical and social protocols, he manipulates context to address larger—and perhaps universally humanistic—themes. In

James, then, and in many other examples throughout the canon, we find literary merit emerging from context.

In this way, one must understand something about context to determine literary merit. Associating the concepts of context and literary merit opens a way for us to frame a question that will allow us to explore how algorithmic tools can cognize context in the coming sections. The question is as follows: can an algorithm determine what texts are "good" in a vast corpus? In order to reduce this problem, and to avoid becoming entangled in the centuries of work on merit and canonicity that extend back to Aristotle, we must also reduce the notion of merit, and clearly define concrete features of what might constitute a good text. To aid in this reduction, I will locate our inquiry in the relatively new, but already densely populated genre of fan fiction.

I propose that identifying good fan fiction is potentially more tractable than locating those transcendent (but largely ambiguous) traits normally associated with academically canonized texts. To put it simply, I believe that many readers of fan fiction would consider a work good if the text reproduces the fictional world of the source material with some fidelity, and furthermore expands upon this world in an interesting and coherent manner. Since fan fiction is inherently derivative, we have not only a readymade context, but a context that is primarily textual in nature. As such, we can factor out many of the problematics inherent in figuring what constitutes historical or social context as a humanistic notion, and proceed, as we did in the previous chapter, by approaching the determination of merit in fan fiction first as a *technical* problem. We will, in other words, attempt to figure context and associated merit in fan fiction as pure information. To aid in this analysis, we will invoke Reading Algorithms in the

second sense I mentioned in the introduction, by imagining what it is like to read like a machine. These thought experiments will yield metaphors that we can use to think about what might be missing in these mechanistic forms of reading. As our first chapter employed a similar analysis to define higher and lower forms of sequencing, our consideration here as to how algorithms must represent and cognize context to answer related questions of merit will bring us to an understanding of how there may exist similar, qualitatively different forms of context.

Section 2.2: The Place of DH Within the Great Unread of the Fan Fiction Universe

We will begin our investigation into the interrelated concepts of merit and context in fan fiction with a brief methodological intervention. Fan fiction remains largely outside the traditional bounds of current academic literary studies.[1] Wherever one situates works of fan fiction on the spectrum of texts that deserve closer interpretive study, there is no denying that fan fiction as a form presents a rich set of working examples to explore more traditional disciplinary questions. Perhaps modern genre theory could benefit from the unique

[1] It should be said up-front that there have been academic studies of fan fiction dating back nearly a decade. Some examples: Busse, Kristina, and Karen Hellekson. *Fan Fiction and Fan Communities in the Age of the Internet: New Essays*. McFarland & Company, Inc., 2006. Busse, Kristina, and Karen Hellekson. *The Fan Fiction Studies Reader*. University of Iowa Press, 2014. In particular, Abagail De Kosnik, an associate professor of New Media at Berkeley, has studied fan fiction in the context of DH. Despite these (and other) examples, it should also be noted that the academic study of fan fiction is still new enough to cause controversy. See, for example, the backlash against a student-led course on fan fiction taught at Berkeley just three years ago: Baker-Whitelaw, Gavia. "What Not to Do When Teaching a Class about Fanfiction." *The Daily Dot*, 11 Dec. 2015, www.dailydot.com/irl/berkeley-fanfiction-class-backlash/. The main contention against the course itself seemed to center around inviting class members to criticize online works of fan fiction. But it is difficult not to detect a certain general opinion in the online commentary, here and elsewhere, that fan fiction should be somehow "off limits" to professional academic literary study. I do understand and sympathize with much of what is said in this commentary, at least as far as the discussion around privacy concerns, and as I will note more extensively in the subsequent pages, I will generally refrain, insofar as is possible, in citing works of fan fiction as one would reference traditionally published literary or academic texts.

manifestation of the "anxiety of influence" that exists within fan fiction, given that these texts are produced as explicit extensions of existing works. Or maybe fan fiction's place among new media and technology – the fact that nearly all fan fiction is published electronically, and draws upon cinema, graphic novels, internet phenomena, and so forth – opens avenues to explore changes in the production and consumption of contemporary literary texts.

But the sheer size of the corpus presents a substantial impediment. Facing a mountain of texts in uncharted territory, most of us would begin by asking: "What should I read?" Which of these texts are important; which ones will give me the best foundation to not only evaluate other texts, but also to understand what, if anything, distinguishes fan fiction from other forms? Can we approach fan fiction using the same tools and training that we use to interpret and evaluate canonical literary texts? If so, what texts are most emblematic of these differences? What texts would I need to read in order to engage with people who have a deep investment in fan fiction, either as non-professional aficionados who read these texts for pleasure, or other academics who are circling around the same initial questions that incited my interest? Which texts should I read?

The majority of publicly available fan fiction exists on a single website: fanfiction.net. There are over six *million* individual works available on this site as of March 2018.[2] To better help us grasp our situation—our question of "What should I read?"—let us imagine that this

[2] Contrast the six million works in the fan fiction corpus with two of the most widely cited collections used in modern DH studies, the ~11,000 copyrighted novels in the HathiTrust Digital Library (https://www.hathitrust.org/) and the ~13,000 copyrighted novels in the Chicago Corpus. Considering the fact that even the millions of works in the fan fiction corpus represent an extremely small niche in the enormous universe of textual data, I often remain skeptical that DH has any claim, relatively speaking, that it is somehow representing the "great unread" to any meaningful extent beyond those texts that fall under the purview of traditional literary studies.

website is a traditional library, and the six million digital works of fan fiction are published as individual books. Picture, then, standing in a space roughly one fifth of the size of the Library of Congress, staring at about one hundred and fifty *miles* of bookshelves. The Library of Congress employs a staff of over three thousand individuals, many of which could give a good answer to the question: "What should I read?" But we are alone in our fan fiction library, and we can wander these imaginary stacks for days and days without coming any nearer to finding the best texts to launch our inquiry. If we want help, we are going to need to create our own librarian.

While fan fiction represents an intrinsically rich case-study from a literary critical perspective, my choice to locate our inquiry in fan fiction is not entirely motivated by a desire to explore fan fiction *per se*. Fan fiction, taken in aggregate, embodies two important qualities: it is both massive and, for the most part, remains largely untheorized from a formal academic perspective. And fan fiction may not be something that can be completely mapped onto existing literary methodologies. If traditional methods of approaching new bodies of literature might not be appropriate for opening inquiries into fan fiction, and, furthermore, if the main impediments in gaining some form of theoretical traction are primarily related to the size of the corpus, then we are moving squarely into the realm of DH. If fan fiction represents a tangible and readily accessible manifestation of the "great unread," then the tools available to the DH practitioner might offer an avenue into this massive corpus. But, very importantly, the question I have chosen to ask of this corpus—What should I read?—is manifestly *not* one DH is prepared to answer. And the reason why DH tools cannot tell you what to read—which texts are good or important—is precisely tied to algorithms' limited access to context.

To begin to understand these algorithmic limitations, it is useful to first explore how DH tools are commonly situated within current academic literary studies. Recently, some prominent DH practitioners have suggested that the function of DH within literary studies is to generate models.[3] Models have existed in literary studies for a long time, circulating mostly as interpretive frameworks. A postcolonial interpretive framework developed around a subset of Afro-Caribbean literature is a model, as is a psychoanalytic methodology of reading 19th century Gothic novels. A model simply shrinks a body of information by transposing it onto a necessary smaller representation. A map is a model of physical space, and just as certain different maps highlight different features of real topography—a road map contains markers that delineate cities and counties, whereas a geological map might omit this information and concentrate on the elevation and makeup of the physical surface itself. Similarly, interpretive frameworks-as-models highlight certain features of the textual subset they hope to explain. And to reiterate the familiar maxim that "all models are wrong," models, whether maps or interpretive frameworks, always leave something out: they are subsets of an informational whole, representing while simultaneously condensing.

The view of DH-as-model-generator implies that the potential interpretive capacity of DH tools emerges *as a result of* the models discovered in the process of large-scale data analysis. Yet this analysis—or more precisely the tools that carry out this analysis—require that the data be organized into a form that is cognizable by the algorithm. As we saw in the first chapter, this organization of things into informational units requires both a reduction into

[3] See, for example, two articles published in the Fall 2017 PMLA retrospective on Moretti's seminal DH anthology, *Distant Reading* (2013) : So, Richard Jean. "'All Models Are Wrong.'" *PMLA*, vol. 132, no. 3, 2017, pp. 668–673. Piper, Andrew. "Think Small: On Literary Modeling." *PMLA*, vol. 132, no. 3, 2017, pp. 651–658.

homogenous units and the anterior notion that the organization *means* something to the analysis at hand. To put it more simply, in most forms of current DH analysis, a human interlocutor curates the corpus and componentizes the data in a manner that can be consumed by the algorithm. This organization of information is itself a form of interpretation. The curation of datasets and the choice of how to represent this data to the algorithm cannot be wholly divested of subjective notions of what might be important to an interpretive model.

What I am trying to draw out here is twofold. First, this notion of DH-as-model-generator is not too far away from traditional literary criticism. In DH there is a cognitive intermediary choosing what data is relevant and the choice of algorithms to produce that data, which is equitable to the cognitive process by which human interpreters determine which passages in a text merit close reading. And, beyond the cognitive process in the choice of data sets and algorithms, the fact that DH information cannot directly yield something like textual meaning, we are also eventually drawn into the more traditional intermediation between textual data and interpretation, a process that all would agree involves some form of conscious cognition. DH practitioners operating under this model-building rubric might argue that their starting evidence is more objective—statistical aggregations as opposed to "exemplary reading[s] of exemplary passages."[4] But by their own admission this aggregation of results

[4] Fleming, Paul. "Tragedy, for Example: Distant Reading and Exemplary Reading (Moretti)." *New Literary History*, vol. 48 no. 3, 2017, pp. 437-455.

cannot be generative of *meaning* until they are transformed by a trained literary interpreter into the assemblage that would constitute a model.[5]

Second, coming back to our focus in this chapter, another way of describing a literary model is as a form of context. An interpretive framework situates writing within a set of contextual expectations. If, as I suggested above, DH models cannot wholly constitute the entirety of a model-as-context, and require human intervention, then it seems provocative to ask what, exactly, constitutes these human cognitive components within idealizations of context. It seems as if DH tools can provide some measure of context, but the context required to construct an interpretive framework contains *something more* than what is provided by algorithmic analysis. DH is uniquely positioned to explore this excess—this difference between context figured as an aggregation of statistical results and the notion of a higher-order context arising from human cognitive assessments of the interrelatedness among heterogeneous concepts, things and perceptions. Yet, despite this potential, most DH studies to date have been focused on generating statistical results and mobilizing these results as evidence toward

[5] Recently, in 2019 and beyond, the view of DH as a *supplement to* as opposed to a *replacement for* of traditional literary studies has gained much traction. Much credit, I think, should be given to Nan Z. Da's influential and provocative critiques in *Critical Inquiry* and the *Chronicle of Higher Education*: Da, Nan Z. "The Digital Humanities Debacle." *The Chronicle of Higher Education*, March 27, 2019. https://www.chronicle.com/article/The-Digital-Humanities-Debacle/245986 and Da, Nan Z. "The Computational Case against Computational Literary Studies." *Critical Inquiry* 45, no. 3 (2019): 601–39. https://doi.org/10.1086/702594. In response to Da's critiques, one finds even the staunchest proponents of DH, such as Ted Underwood, mapping out spaces where statistical models coexist next to traditional humanistic approaches to interpretation: "If, instead, humanists welcome new forms of curiosity about the past, and work to create a curricular foundation for them, we might be able to build new bridges between our departments and other corners of campus. If we succeed, new forms of research will benefit from humanistic expertise. Students will learn that the humanities are not a moral sanctum set apart from the world, but a mode of inquiry closely connected to other parts of their intellectual lives, including the pleasure of building models and solving problems." Underwood, Ted. "Dear Humanists: Fear Not the Digital Revolution." *The Chronicle of Higher Education*, March 27, 2019. https://www.chronicle.com/article/Dear-Humanists-Fear-Not-the/245987.

traditional literary-critical claims. In this chapter and throughout this work, I am more interested in where this arc *fails*, for these point to spaces where we can examine in detail qualitative differences between algorithmic and humans' cognitive engagements with texts. In this alternate vision of DH, we readily accept the failures of algorithmic tools to fully encompass the desired results, and explore the limitations exposed by these failures as possible spaces of higher order cognition.

With this methodological intervention in mind, I will structure the subsequent sections exploring literary merit and context in fan fiction as a series of experiments that foreground different forms of algorithmically derived context. In the true spirit of experimentation, I begin with the supposition that my analysis might fail, but with the conviction that these failures will yield valuable information not only about the abstract contexts of context and merit, but also insights as to how these algorithmic tools represent and cognize their informational universe.

Section 2.3: Literary Merit as Preference

With the above methodological goals in hand, let us step into our laboratory and begin our experimentation. Imagine this lab situated within a vast library containing most of the extant works of publicly available fan fiction. We stand staring at the miles of virtual shelves and ask where to begin. As mentioned previously, we are fortunate in that most published fan fiction is collated on a single website, fanfiction.net. Browsing to the base URL (see Figure 3), one will notice that the greeting page contains links to the type of source material (e.g., books, graphic novels, TV, etc.).

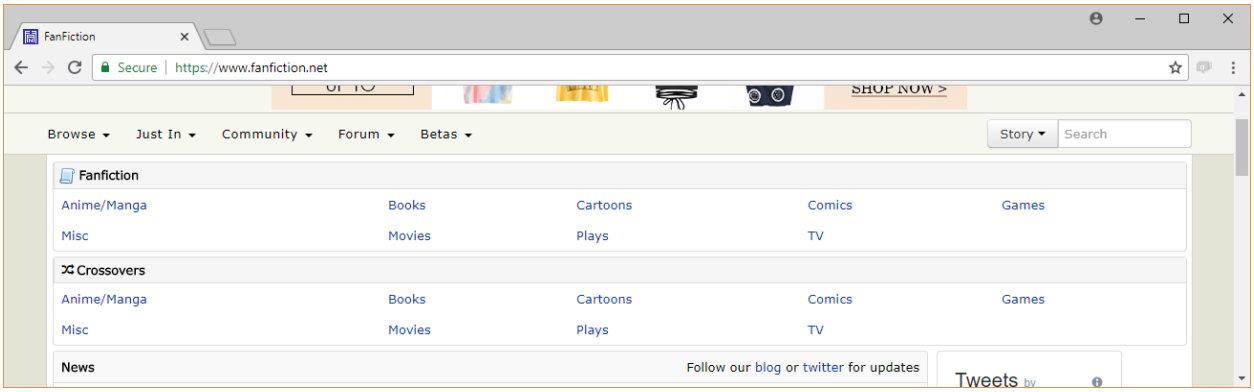

FIGURE 3. FRONT PAGE OF FANFICTION.NET

Clicking on “books” will bring one to a very long page of links, each of which lists the novel or series from which the fan fiction is derived, along with the number of individual texts in that category.

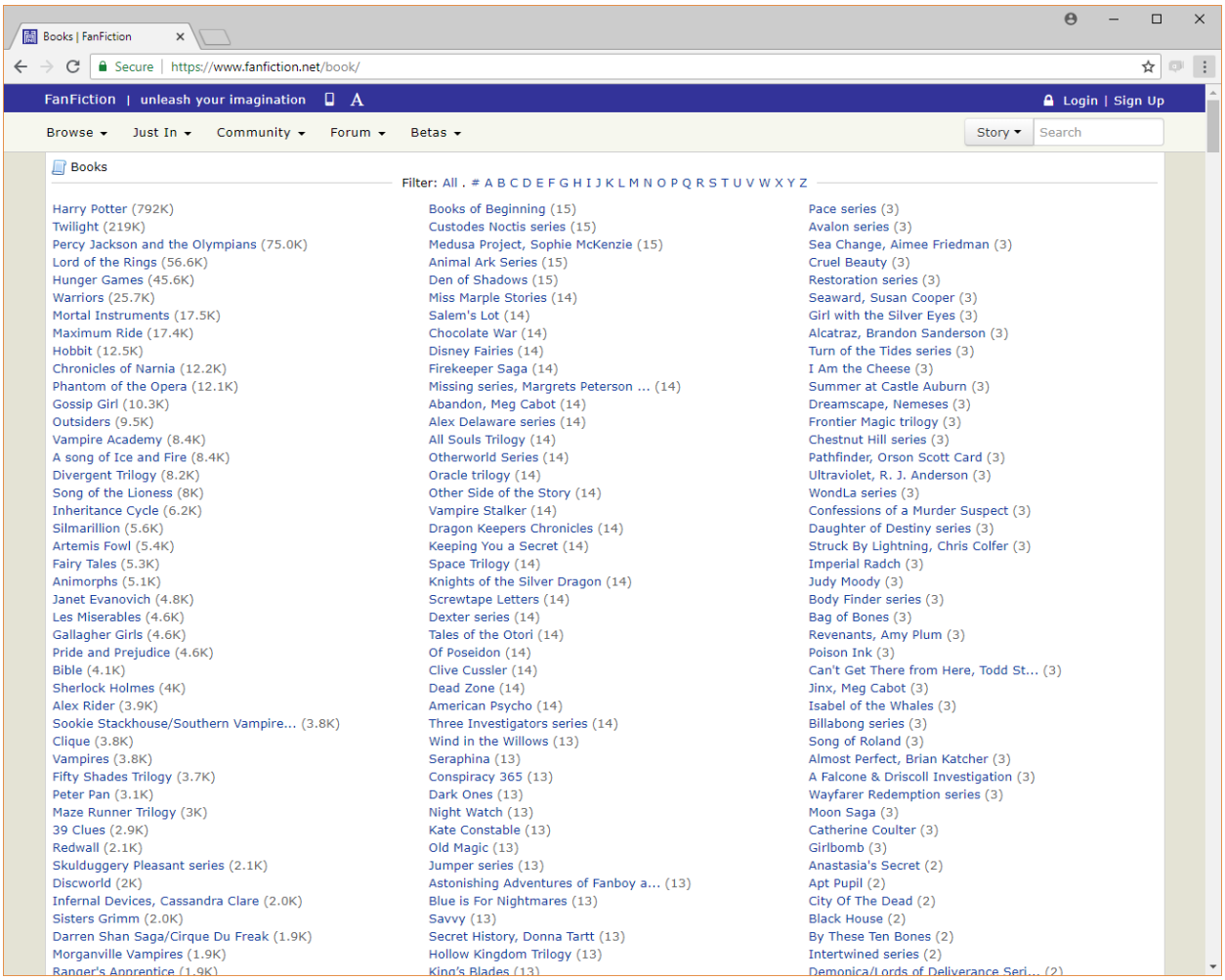

FIGURE 4. LIST OF BOOK-BASED FAN FICTION, CATEGORIZED BY SOURCE, SORTED BY NUMBER OF WORKS

What is immediately striking is the sheer number of texts. To take the two highest categories, there are over one *million* fan fiction texts based on *Harry Potter* and *Twilight* alone. And there are hundreds of source books. The overall distribution of fan fiction texts is top-weighted toward five source texts: *Harry Potter* (792k), *Twilight* (219k), *Percy Jackson and the Olympians* (75k), *Lord of the Rings* (56.6k) and *Hunger Games* (45.6k). There are at least two common threads among these top five novels: they are—apart from the possible exception of *Lord of the Rings*—what would now be termed "young adult" fiction, and all of them have had

movie adaptations. But alongside the young adult novels, there are "serious" source texts represented on this list as well: *Pride and Prejudice* has 4.5 thousand fan fiction adaptations, *Moby Dick* has twenty-seven, and *Middlesex* has two.

Drilling down into each category is, as one can expect, immensely fun, and reveals works that span the literary map in both form and content. To take a random sampling, one of the entries on the fan fiction collection for *Moby Dick* contains what appear to be song lyrics[6] :

[1]

CHORUS
There she blows
The fight is about to start
And her body is shaped
Wider than the sharks
There she blows
Huge machine of war
Ready to make you her prey
She makes them boat cp'tain say
There she blows

[2]

Well done, Ishmael, I'll survive
Yeah, the Way Her Body Moved Like A Hurricane
Thought I Knew What Deadly Was But Uh... Uh She Just Charged

[6] It should be noted that the authors of fan fiction content are identified by a simple "handle," an alias that may or may not be related to the "real life" author. While fan fiction is public domain, and information about specific authors is controlled by the authors themselves, I will refrain from referencing specific texts by name or authors by handle as a matter of general privacy. It is likely that at least some fan fiction authors are minors, and, as opposed to professional authors of traditionally published texts, it is also likely that many fan fiction writers are not completely aware the mechanics of citations and copyrights. It therefore feels somewhat off-putting to directly cite and reference these works (with attributions) within an academic publication.

She's A Danger, Monster
Fighter, Killer
She Knows Exactly What She's Killing, I Think She's A Pro
When She swim in The Sea She Make Sure That We Know
She's A Thriller, Killer
Huge sea monster
(Chorus)…

These lyrics recall the exuberance of Pynchon, and it is simply fun to imagine these put to music. Each of these avenues contains similar gems and surprises, making it tempting to spend days wandering this corpus aimlessly. The temptation to wander these virtual "stacks" is one that, I confess, couldn't personally resist at several points during this study. But when surfacing from such an expedition, I was always bought back to that page listing the hundreds of source texts (fig. 3) and the thousands of fan fiction texts derived from each source, and the attendant realization that that front page, containing millions of aggregated listings, only represents those fan fiction texts derived from *novels*. There are entire, other pages listing fan fiction derived from TV shows: more than 25 thousand works derived from the various *Star Trek* series, and 75 thousand derived from *Dr. Who,* for example. All of this, taken in its dizzying aggregation, impresses the enormity of the problem we are up against. In this uncharted vastness, what should I read? Which ones are good?

To get a foothold on this question, I began my DH experimentation by writing a mechanism to aggregate and download all the texts from the fan fiction website. These programs are called "web crawlers" and they represent a common automated methodology to aggregate information distributed across (in our case, millions) of individual HTML pages. A

web crawler starts with a given URL and then automatically traverses each link on the page, just as a human might by clicking on each of the links. The web page data is then saved to a local storage, such as a hard drive, and the crawler continues its work. As part of its operation, a crawler can often identify "meta-information" on each of the pages it visits, as well as "clean" the main article, resulting in simple, easy to work with text files, each representing an individual work of fan fiction. What results is not unlike a library indexed by a card catalog. In the case of the fan fiction site, I was able to collect about 6 million articles in total, each indexed by the following fields:

1. Title
2. Author
3. Category (e.g., *Pirates of the Caribbean*)
4. Genre (e.g., Adventure, Humor)
5. Language
6. Status (e.g., Completed, In-Progress)
7. Published, Updated, Packaged (all in date format)
8. Rating (K+)[7]

[7] In the fanfiction.net rating system, there are the following suggested ratings: *K* (general audience, 5 years or older), *K+* (mature children 9 years or older), **T** (teens, 13 or older), *M* (mature, not suitable for children younger than 16) and *MA* (content suitable only for mature adults). Fan fiction has specific sub-genres that tend toward the pornographic, such as "slash fiction," which imagines same-sex relationships between characters in the source material. While not all slash fiction is pornographic by any means, some of it can be, and the graphic texts will garner a "MA" rating. Fanfiction.net, as a site, disallows the posting of MA material, but some MA material does make its way into the selections, as there is just too much content to moderate, and too few volunteer moderators. In the context of our DH inquiry, this inability to moderate MA context recalls Supreme Court Justice Potter Stewart's famously worded opinion in the 1964 *Jacobellis v. Ohio* case regarding "hardcore" pornography: "I shall not today attempt further to define the kinds of material I understand to be embraced within that shorthand description; and perhaps I could never succeed in intelligibly doing so. *But I know it when I see it*…" [emphasis mine]. One wonders if DH tools could identify "hardcore" pornography, however one wishes to constitute that category. I personally do not think so. And if my intuition is correct, one wonders whether Justice Stewart's opinion is stating that higher-order cognition is implicated in what he terms "seeing." And to intentionally complicate this further, his opinion seems to indicate that he "cannot articulate" what in our context would constitute an MA rating. When one is dealing with *linguistic*-based material, such as these fan-fiction texts, one is certainly tempted to ask why the *thing* (i.e., hardcore pornography) can be articulated but not its *characteristics*. We cannot rule out simple sensibilities of decorum in this instance, but we seem to have an example of a *feature* that, for whatever reason cannot be reduced to *phrases*, even if that feature always manifests itself as strings of phrases. The fact that this quote is still in circulation suggests, I think, that the concept of inarticulable quiddity—of knowing only by *seeing*—still refers to a cognitive capacity that is mysterious *qua* its extralinguistic character.

9. Number of Words
10. Summary (free text, e.g., "Izzy, Bella, Robbie and Edward go on a cruise on the black pearl")

In addition to the above categories, which were, from a technical perspective, relatively easy to gather, I added a more elusive category to our "card catalog": *number of reviews*. Each fan fiction text was crosslinked to another site where readers could post reviews, a Yelp-type site dedicated to each text, complete with star ratings and space for free text commentary.

While technically challenging, the "number of reviews" statistic was one that I believed could allow us to explore certain facet of merit – a simple measure of a texts' popularity. Once collected, the first overarching impression I had was the immense readership devoted to fan fiction. For example, the most-reviewed story on fanfiction.net, a 101-chapter, 561-thousand-word work, titled *Harry Potter and the Methods of Rationality*, had nearly 34 *thousand* reviews.[8] Every statistic here drives home the enormity of this corpus: fan fiction authors are regularly producing texts in excess of one thousand non-virtual pages, and their collective readership—even estimated solely by the number of individuals who actively post comments—is often in the tens of thousands. The readership of *Harry Potter and the Methods of Rationality* is likely on par with a modern best-selling, "serious" author—Dave Eggers, say—whose *A Heartbreaking Work of Staggering Genius* (2000) has sold 400 thousand copies, and garnered eight thousand reviews on Goodreads. In our fan fiction corpus, there are over eight thousand works with over a thousand reviews, and forty with over ten thousand. There are nearly a thousand fan fiction works longer than five hundred thousand words. In short, in fan

[8] *Harry Potter and the Methods of Rationality* was the subject of an entire section in the ill-fated 2015 Berkeley fan fiction undergraduate course mentioned in the opening note at the beginning of this chapter, making it, perhaps, something approaching a "canonical" work of fan fiction.

fiction we are dealing with a corpus that is at least commensurate to, and likely more extensive than, the whole of modern "serious" literary output. We again circle back to a variant on our original question: is this enormous body of fan fiction also commensurate in its level of seriousness? If so, which ones should I read first; which of these millions of texts are worth reading to arrive at an assessment of the corpus' merit as a whole? And is a text worth reading simply because it is widely read and reviewed?

Phrasing the above question in terms of "worth" shifts us, at least temporarily, onto a ground that is less aesthetic and more economic. The inherent dissonance between these vantages will yield a set of problems that will move us closer toward our core issue. Consider, for instance, how a company like Amazon might approach the question of "what is worth reading?" Amazon, alongside thousands of other entities, perform this calculation every day. These calculations generate recommendations for us, automated "suggestions" of what a text might be worth. Very importantly, the "worth" under discussion here is not, by any means, *intrinsic* worth; it is rather the contemplation of "what is this text worth *to me*." Although it may not be immediately obvious, if you spend any time at all browsing the internet, this calculation—and it is very much a calculation—is being done on your behalf several thousand times a day at speeds well below the threshold of conscious perception.[9] When you browse to a popular internet site, the overwhelming odds are that the content you came to view is bracketed on either side by banner advertisements. The slight delay that you might perceive in

[9] These calculations, occurring below the threshold of conscious perception (normally taken as 500 milliseconds) yet which likely influence cognition, form the subject of two influential studies coming out of Duke's New Media department: Hansen, Mark B. N. *Feed-Forward: on the Future of Twenty-First-Century Media*. University of Chicago Press, 2015 and Hayles, N. Katherine. *Unthought: the Power of the Cognitive Nonconscious*. The Univ. of Chicago Press, 2017.

the page "loading" is very likely not due to the speed of your internet connection or the rate at which your browser renders the graphical content. That slight, barely perceptible delay is, in fact, an auction. It is an auction on your attention: a literal bidding war, carried out electronically, based on the information various firms know about you from a variety of sources. Behind the scenes, in timespans measured in the tens of milliseconds, groups of companies are paid to collate information about you and bid against each other for the right to display their ad. One company might know that you are an academic professional and recently viewed a group of texts on Dickens. The algorithmic bidder will calculate that it will pay up to 1/20th of a cent for the right to advertise the latest anthology of Dickens criticism, just released on Amazon. If the company wins the bidding war, and you subsequently click through on that advertisement, the electronic auctioneer may receive one cent, a profit of 2000%. The breadth of information available to these bidding algorithms is staggering and often unnerving. After sending a copy of this dissertation via Google mail to an acquaintance for review, banner advertisements promoting commercial machine learning software and hawking texts and courses on ML programming dominated my browser sidebars within fifteen minutes. It is obvious that the problem of what a thing is worth *to me* has been mapped out with astonishing accuracy in modern digital technology.

But is this calculation of context figured as worth *to me* appropriate to our problem of "what to read" in fan fiction? To answer that question, we need to piece apart the multiple facets of the algorithm. To begin, we need to consider what this calculation implies about the nature of subjectivity. A subject, in the "eyes" of Amazon's preference-related recommendation algorithm, is a *potential* consumer, and therefore the most useful model of

subjectivity for this algorithm simply reduces you-as-subject to a probabilistic outcome: will this subject buy this recommended book or not? Subjectivity, to this algorithm, is therefore synonymous with probability. Individuals are simply aggregations of informational traits that, when taken together, form the basis for the computation of "what is the likelihood that this subject will do X?" X in the above case of the banner advertisements is the action of clicking though to the seller's website, and in the case of fan fiction, X would be "what is the probability that you will *like* this particular text." In this model, the only traits that are relevant to the makeup of the digital subject are those that have a direct bearing on the calculation of the probability of the potential action or emotional state.

Notice how the X in the above formula—the actual object under contemplation—fades from the picture. X, from the point of view of this simplified algorithm, doesn't *really matter*; what matters here, from the algorithm's viewpoint, is entirely contained within the subject. The worth of X is precisely the worth of X to the *subject*. And this is the first issue with conflating the question of "what *should* I read" with "what would I *like* to read." A thing's value, taken in isolation and phrased in economic terms, is directly related to its worth to me as an individual. A fur coat has no value to me; I would not buy one for any price for ethical reasons personal to me. Yet another individual might pay thousands for the same item. This is simply to state that software algorithms, such as the code driving the price that determines the value of a banner ad to me, are operating in a different domain than a cognitive agent that would determine some form of worth that *belongs* to the object. "What is this book worth to you?" is a question located in the subject; "What is this book worth?" edges into an objective

territory that will very much become central to this chapter, but at this moment in our discussion, is not at all clear.

Given the limitations of determining what a subject might like based entirely on traits of the subject, it is not surprising that companies such as Amazon have developed more sophisticated algorithms to determine consumer preferences. One way of enhancing the efficacy of a preference algorithm is to determine what other, similar customers like and then mapping their preferences onto your own profile. In this approach, the algorithm is predicated on the ability to determine what constitutes a "similar" customer. As one can imagine, a company like Amazon stores many pieces of information about a customer: their age, gender, geographical location, and so forth. This is where ML comes into play. Some of these customer attributes are more relevant than others, and numerically, calculating which attributes contribute most to your decision can represent a daunting problem. But part of the "magic" in ML algorithms is that they can start with a result and work backward, in effect, to find the attributes that contribute most to that result. In our example, to figure out the most important characteristics of readers who like a given fan fiction text, we'd train our algorithm by inputting the set of all readers, in its entirety, and then giving the ML algorithm positive feedback when it guesses correctly that a reader likes that text. In this manner, the algorithm might eventually observe that most readers who like *Harry Potter and the Methods of Rationality* identify as male, are between a certain age bracket, tend to live in urban areas, etc.

But what have we done here? We have developed an ML algorithm with the ability to predict whether a reader might like a *given* text based on a certain idea of what constitutes a "readerly" subject. There is a strand of apriorism running in two directions here. First, we must

have already built a data set containing the preferences of many readers. But furthermore, we must also determine ahead of time the criteria that constitutes a subject. If we are not careful, the criteria could be heavily weighted toward the end-result that we are trying to determine. We could very well include the criteria "likes the original *Harry Potter* series" into our definition of a subject, and one would imagine we'd find that there'd be a nearly 100% correlation between the trait "enjoys the original *Harry Potter* series" and those subjects that enjoy *Harry Potter and the Methods of Rationality*.

There is an insidious aspect to this formulation of worth rooted in reductive contexts constructed around notions of collective preferences of similar individuals. ML algorithms employed in this capacity often build subjects based on subsets of criteria determined *before* the algorithm runs. This informational reduction, familiar to us from our discussion in chapter one, can result in an immensely vicious circularity, a self-reinforcing process that is particularly alarming because it is not yet widely acknowledged within most technical communities. Take an example taken from one of the most prominent, recently published technical Machine Learning texts, written by a programmer at Google involved in the design of their cutting-edge ML toolkit (the now-open source software known collectively as *TensorFlow*). This extremely technical book, written in 2017 by Aurélien Géron, contains a discussion on "precision" versus "recall" in ML-based image recognition applications. In that discussion we encounter the following observation:

> For example, if you trained a classifier to detect videos that are safe for kids, you would probably prefer a classifier that rejects many good videos (low recall) but keeps only safe ones (high precision), rather than a classifier that has a much higher recall but lets a few really bad videos show up in your product (in such cases, you may even want to add a human pipeline to check the classifier's video selection). On the other hand, suppose

> you train a classifier to detect shoplifters on surveillance images: it is probably fine if your classifier has only 30% precision as long as it has 99% recall (sure, the security guards will get a few false alerts, but almost all shoplifters will get caught). [10]

Géron is essentially stating here that it is unthinkable for children to be exposed to "unsafe" videos—which is, admittedly, not something anyone wants to happen—and he recommends that decisions made by these ML algorithms should be thoroughly checked, even by humans if necessary, to ensure high precision (i.e., no false positives). On the other hand, it is "probably fine" that security guards get a few "false alerts" when the ML system misrecognizes a shoplifter, because "almost all shoplifters will get caught." One can assume that Mr. Géron has never been handcuffed in front of his family on a routine shopping outing and subsequently forced to spend the night in a community holding cell as a result of an ML determination that he belongs to a set of "probable shoplifters." In the face of such monumental disregard for the potential consequences of ML "misclassifications" as displayed by Mr. Géron—one of the leading ML researchers employed by one the world's most prestigious companies—we have a glaring example of the flaws arising from the algorithmic construction of contexts as artefacts of putative similarity among individuals, especially when that context is deployed as a probabilistic measure of subjective intention or identity.[11]

---

[10]Géron, Aurélien, and Rebecca Demarest. *Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly, 2017 (Kindle Locations 2037-2039).

[11] This is not an isolated incident. Recently another group, operating out of Stanford, for some reason decided that it would be important for an ML-based facial recognition algorithm to determine sexual orientation from facial features. While perhaps, to some, not as odious on the surface as Géron's example, the overwhelming question that comes to mind in this particular study is "why?" Of all the things ML can be used for, why choose this? Again, one would think an institution such as Stanford might know better. See: Wang, Yilun, and Michal Kosinski. "Deep Neural Networks Are More Accurate than Humans at Detecting Sexual Orientation from Facial Images." *Open Science Framework*, 16 Oct. 2017.

Coming back to our analysis of an Amazon-type preference analyzer, where we have a ML-based algorithm generating the familiar "other customers like you have purchased…" recommendations, we can now see how this type of process could also become circular. Because the algorithm makes it very easy for me to "like" the same items that have been recommended to me (they are, after all, just a click away) the predictions made by the ML tool will eventually become self-reinforcing. Ironically then, very poor choices of the traits that would define reading subjects – i.e., choices of subjective traits that would result very high, nearly tautological, correlations between readers and subsets of books, such as our aforementioned example of "subject enjoys the original *Harry Potter* series"—would be those that would be most generative of further "likes." In this way, these very sets of overwhelmingly correlated traits, which, from a statistical standpoint are terrible choices to delineate meaningful subsets, can become exactly those traits that will emerge to define groups of readerly subjects.

Despite these potential pitfalls, these more sophisticated "preference" algorithms take us one step beyond the question of "how much is this text worth *to me*?" By incorporating the preferences of others into the algorithm, we have shifted our stance slightly, asking, in effect: "how much is this text worth to others *like me*?" Transposing this analysis onto literary-critical grounds, we seem to be edging toward a slightly more familiar metric of literary merit. We still are entirely rooted in *subjects*: we have not considered, in any way, the text itself. And we have seen the dangers of constituting subjects as aggregations of discrete traits, which algorithms must do in some form or another. Yet despite these caveats, let us remain in the realm of literary criticism, and suppose that we simply insist that our subject consists of only one trait:

they are a trained literary interpreter. In this system, it is nonsensical to talk about the inherent *value* of a text, or at a more fundamental level, the ontological status of text as a thing that is generative of meanings. Textual value and interpretive activity are inseparable from the *readers* who retroactively convey that meaning. We have arrived, albeit along a somewhat circuitous route, at the central thesis of Stanley Fish's 1982 work, *Is There a Text In this Class*, which finds him, positioned at the very inception of the upcoming theory wars, wrestling with the notion of absolute interpretive relativism:

> At this point it looks as if the text is about to be dislodged as a center of authority in favor of the reader whose interpretive strategies make it; but I forestall this conclusion by arguing that the strategies in question are not his in the sense that would make him an independent agent. Rather, they proceed not from him but from the interpretive community of which he is a member…it is interpretive communities, rather than either the text or the reader, that produce meaning and are responsible for the emergence of formal features.[12]

What we have drawn out, then, is the equivalence of a currently widespread algorithmic method of assessing textual value with an admittedly dated—but initially well-received and perhaps still relevant—theory of interpretation. Fish's argument aligns closely with a central property of nearly all iterations of Machine Learning algorithms, namely that *consensus* determines *classification*. What a thing *is*, at an ontological level, is beside the point. What matters is what a thing is *determined to be*, a determination that is necessarily based on a statistical aggregate of agents, which may operate autonomously but only achieve agency when taken as a collective. To become a part of a collective these agents must necessarily be classified themselves: in Fish's view, these agents must be members of an interpretive

[12] Fish, Stanley. *Is There a Text in This Class?* Harvard Univ. Press., 1982. p 13-14.

community; in our example of Amazon's ML-based preference algorithm, these agents must be *like me*, a determination that likely has its roots in a demographic-based determination of subjectivity.

A potentially surprising result of approaching literary critical theory from an algorithmic perspective is that *preferences can be determined without any knowledge of the object*. What I like, in other words, can be derived entirely from my inclusion within a group of similar subjects, and when one presses hard on Fish's notion of interpretive communities, those communities are, likewise, in no way constituted by properties of the objects (texts) that they study. To employ a Derridean concept, there must exist an *arche*-preference, or an *arche*-interpretive community that speciously confers the appearance of legitimate inclusivity, which allows us to proceed at a rhetorical level without considering how these categories might be constituted. What matters insofar as we speak of literary worth, is, in this line of reasoning, my inclusion within a group of people *like me* or an interpretive community *to which I belong*. We have demonstrated this curious result practically: an Amazon preference analysis algorithm does not at all care what I'm interested *in*; the particular objects I *like* are fully derivable from the desires of people *like me*. Furthermore, the information about subjects that drive this algorithm was collected long before the algorithm was written. And finally, the categories of subjects that drive the preferences were delineated after the initial, generalized data of the subjects was gathered. What the algorithm thinks we like is therefore determined not only by the preferences of others, but these "others" only exist as such to serve to delineate preferences; they come into being out of an initial dataset of characteristics that was gathered long before my own preferences came into consideration.

Section 2.4: The Ambient Context of Phrases

In the last section, we saw how literary merit constituted as literary preference can be algorithmically determined without any consideration of the texts themselves. Amazon's automated preference tools guide you through their vast corpus by considering what you would *like* to read, a determination that can be algorithmically determined within the context of other readers. But most concepts of literary merit are not explicitly based on what you would like to read, but rather, in some form or another, what you *should* read. Criticism encompasses a judgmental dimension that considers characteristics of the text itself. As I have suggested, this judgement very often involves an assessment of the efficacy of a text's engagement with its context. In traditional literary criticism, what constitutes this context is multifaceted and would, I think, be too elusive for our algorithmic experimentation here. But within fan fiction, we may be able to consider the source texts themselves as the primary context informing the determination of merit. A fan fiction work in the *Harry Potter* universe is good, at least in part, if it maintains some fidelity to the contextual world of the original works and coherently expands that world. These are obviously not the only criteria that can contribute to merit in fan fiction, but they are attributes that both seem, at first glance, to reside within the textual data itself and are potentially recognizable by algorithmic analysis.

Our first experiment then, is to explore what form of "contextual world" can be derived from a corpus of texts taken in and of themselves. As a first approximation to the complexity inherent to this problem, let us consider one concrete example of one of the simplest DH analytic measures in contrast with a common literary-analytic methodology. Take, as our

working example, one of the most fundamental tasks in many DH analyses: simply counting occurrences of words across sets of textual data. Although word counting seems straightforward, there exists much variance among word counting algorithms. Some counting methods employ "lemmatization"—a methodology that considers different forms of the same base word (e.g., "work," "working," and "works")—to be identical. As a result, a lemmatizing algorithm will condense all forms of the base word (the lemma) into a single count, which will, in this example, increase the number of occurrences of the lemma "work," while entirely discounting instances of the derivations "working" and "works."

This type of bucketing may be appropriate for some analyses. But from a humanist standpoint, consider how deeply these lemmatizations can come into play in the process of close reading.[13] I recall, for example, a discussion that occupied most of a class period, contemplating how Henry James used the phrase "I was *struck*," as opposed to his more common locution "It was *striking*." The close reading was heavily informed by a form of word counting: when we considered other examples in the text where "struck" and "striking" occurred, we became convinced that this instance was atypical, and thus was important to our interpretation. In close reading, we consider a form of local complexity, which functions by effectively reducing the informational set under analysis so anomalies that are spatially contained within the same text become apparent as locations for further analysis. Algorithmic

[13] In a DH study of gender dynamics in early 20th century book reviews in the *New York Times*, Matthew Lavin delves deeply into the impact of lemmatization in statistical analysis of textual data, giving concrete measurements of how lemmatizing of certain terms under analysis (such as words associated with gendered "virtues" or economic class) can lead to substantial variance in the end results. See Lavin, Matthew J. "Gender Dynamics and Critical Reception: A Study of Early 20th-Century Book Reviews from The New York Times: Published in Journal of Cultural Analytics." *Journal of Cultural Analytics*. January 30, 2020. https://culturalanalytics.org/article/11831.

analyses often gravitate in the opposite direction, exploding the information set across an entire corpus. In DH word counting, then, we deal with locality of phrases in a different manner, as something which emerges statistically, distributions of phrases that are considered only within the context of the informational whole. While DH analysis might be able to infer some type of information from the *repeated* proximity of words or phrases next to one another, it cannot get to the level of close reading. But *why* DH cannot get at close reading is not at all clear. Both close reading and DH analysis have a notion, at one level or another, of what is usual in a set of textual data, and both methodologies can therefore point to locations where we find contextually atypical usages of words or phrases. But DH cannot move to the next level, where close reading considers what this atypicality might *mean.*

While simple-word counting cannot provide a meaningful context to repeated or unexpected phrases, there is something powerful that has emerged from the consideration of distributions of words as giving rise to a certain set of probabilistic expectations. Language, if it could ever be taken in aggregate, is manifestly *not* random from a statistically distributional standpoint. Certain words and certain phrases simply occur next to one another more frequently than others.[14] This nonuniform word distribution, as it manifests itself in language, taken in aggregate or within some working subset, is attributable, at least in part, to grammatical constraints. It is extremely improbable to encounter the phrase "the it the" in the English language, as it is simply inconsistent with grammatically correct sentence structure.

[14] The claim that word usage within natural language is not randomly distributed was put forth as early as 1954 by the linguist Zellig Harris, who will become central to our ensuing discussion of modern NLP techniques. Harris' original essay was republished in "Distributional Structure." *Papers on Syntax*, 1981, pp. 3–22.

This nonrandom word distribution became empirically demonstrable once machine-based natural language processing began to regain traction in the early 2000s and ready access to computationally powerful hardware via readymade toolkits began to increase post 2010.[15] The key development in NLP that demonstrated the consequences of nonrandom word distributions were a set of algorithmic tools that produced models collectively known as "word embeddings." Word embeddings express, in mathematical form, a multidimensional space that shows clusters of words or phrases that repeatedly appear proximate to one another in the textual data.

The computational construction and manipulation of word embeddings yielded a set of algorithms that, quite amazingly, displayed an ability to seemingly cognize sophisticated contextual relationships. To take two examples: one can construct a word embedding from a corpora of texts about transportation, input a word into the model generated by this algorithm—"speed" for example—and the code will return conceptually meaningful terms clustering around that word: "road," "vehicle," and so forth. But far more compelling to me was the ability of these tools to yield solutions to analogies. Inputting "London" is to "England"

[15] The original mathematical framework behind word embeddings is generally attributed to Yoshua Bengio in 2003: Bengio, Y., Ducharme, R., Vincent, P., & Janvin, C. (2003). A Neural Probabilistic Language Model. *The Journal of Machine Learning Research*, 3, 1137–1155, available online: http://www.jmlr.org/papers/volume3/bengio03a/bengio03a.pdf. NLP word embeddings did not gain practical traction, however, until 2013, when Tomas Mikolov published a computationally efficient algorithm in Mikolov, T., Corrado, G., Chen, K., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *Proceedings of the International Conference on Learning Representations* (ICLR 2013), 1–12, available online: https://arxiv.org/pdf/1301.3781.pdf. One thing to note here is the decade-long disparity between the initial theoretical idea and an algorithm that can efficiently implement that idea. This is not uncommon in the practice of Computer Science of any stripe, but in this NLP example, the decade-long gap between an idea and its implementation suggests, at some level, how seriously we must take the idea of *complexity* in language processing. When translating algorithmic processes into humanist terms, we must be extremely careful not to equate *quantitatively* daunting problems with things that seem *qualitatively* beyond the reach of any noncognitive process.

as “Paris” is to X, will find that X to be “France.” This model, in some form, “knows” the semantic relationship between a city and a capital—a relationship that contains many layers of information about *the world as it exists outside of the text*, from the simple distinction between cities and countries to the notions of capitals and governance.

This discovery popularized what is now known in the NLP community as the “distributional hypothesis,” which is the notion that one can derive *meaning* solely from word distributions.[16] But when stated bluntly, as I have here, the distributional hypothesis does not fully account that there could exist different modalities of *meaning*. In other words, when one claims from an analogical demonstration like the above that this algorithm “knows” things about the world, one needs to consider that the only recourse this algorithm has to any form of phenomenological “outside” must be contained wholly within textual data. As human agents within the world, we have radically different access to that “outside,” and however one wants to formulate that access, the key thing here is that this algorithm cannot possibly “know” things in the same way we picture human cognitive knowledge. And yet, the algorithm seems to demonstrate some form of access to higher-order information. Imagining how this algorithm “knows” will therefore give us our first example of what Ian Bogost has so enviably termed “alien phenomenology.” This conceptualization—our first worked example of “alien reading”—

[16] As mentioned previously, the distributional hypothesis was formulated well before modern computational NLP techniques by Zellig Harris in 1954. One finds early forms of this hypothesis trenchantly summarized as “a word is characterized by the company it keeps.” Harris is extremely careful in his invocation of “meaning” in his original paper, but one frequently finds that term circulating in modern discussions of word embedding techniques. Perhaps this is due to the movement of the distributional hypothesis from formal linguistics into the realm of computer science, where “meaning” tends to be invoked in a more colloquial context.

I hope to show how what appears to be higher-order context can emerge from an algorithm, as well as what the limitations of this context might be.

Consider standing in front of a very large library, staring at shelves and shelves of texts. Immediately someone steps in this library and, like William Burroughs, starts cutting up the books into 10-word phrases and dropping these strips of paper in a random pile on the floor. Let's imagine that this scissor-wielding vandal has some preternatural ability to duplicate these books, so we end up with strips of 10-word phrases that overlap: each one starts one word after the last one left off. For example, if one phrase is cut out of a book that starts "The truth about the world is that anything is possible. Had you not seen it all from birth…" you'd end up with many 10-word strips, one which starts "the truth about the world is that anything is possible," the next would read, "truth about the world is that anything is possible had" and so forth. The pile therefore is not only enormous, but you'll also note that you have no idea where one sentence left off and the other begins, nor do you know what book the strip you are looking at was cut from. You do, however, have two things going for you: you possess your own preternatural ability to process these phrases incredibly quickly, as well as the ability to store them in an immense memory.

You have so much memory, in fact, that you decide the best way to make sense of this mess is to take each individual word, say "world" in the above snippet, and put it alongside all of the 10-word strips that also contain the word "world." You also can duplicate the strips yourself, so that you end up with an enormous pile of strips, each based around a single word. In this way, your pile of 10-word strips that corresponds to the word "world" contains *all* of the usages of "world" in any 10-word sequence. Now you have an enormous amount of data,

categorized by word, one of which contains all the words that have *ever* been used within a 10-word context of the word "world."

What can you do with all of this? Let's start with something easy. Suppose someone comes into your now-highly organized library and asks: "what word is most probably going to be used in a phrase that contains the word 'world?'" You are prepared for this: you simply go through your stack of 10-word strips under the "world" label and perform some simple calculations (likely disregarding words such as "and" or "the" – which are sometimes known in NLP as "stopwords"). Your answer might very likely be "big" or "round." Now one degree harder: your visitor asks the answer to an analogy: "Man is to King as Woman is to X." You arrange your three stacks of strips corresponding to "Man," "King" and "Woman" and you go through each, calculating what word is most likely shared between these three stacks. You answer, plausibly, "Queen."

What *kind* of knowledge does this represent?[17] To frame this question, first let's ask a question we *can't* answer given the data we have. The first and perhaps most obvious class of questions we will not be able to answer are questions about the "outside" world. I won't be able to tell you if it is raining outside right now. But that is not strictly a failure of cognition as much as *access*. There exist many noncognitive sensors that can tell you if it is raining outside. But this notion of *access* is important. Can I answer anything about the "outside world" given

[17] Franco Moretti addresses this same question in an article that implicitly outlines relationships derived from a word embedding model. Moretti observes that these models express "Correlations, yes; but no theories, no models, not even explanations." I agree wholeheartedly with Moretti's assessment here, and hope to show, via our exercise in "alien reading," is a more concrete explanation *why* word embeddings cannot capture these elusive traits Moretti invokes. Moretti, Franco. "Patterns and Interpretation." *Stanford Literary Lab Pamphlet 15*, September 2017. https://litlab.stanford.edu/LiteraryLabPamphlet15.pdf.

my strips of paper? Here we must sharpen our question a bit. When we say "outside world" from a literary-critical standpoint, this often refers to some notion of historical context. Now what access do we, as humans, have to historical context? The obstinate answer would be "none," since we have no access to the non-experiential past save for what we *read* or are otherwise *told* via language. While this strong linguistic determinism represents a tenable answer, it might not entirely be true. We do have some notion, in whatever form that may exist, of what it is like to *be now*, and from that can extrapolate to imagine what it might constitute to have *been before*.

The question then becomes: how much of this cognitive extrapolation is rooted in *linguistic difference* alone? The aware interpreter will know that the word "car" has meant very different things in different eras: an automobile, a train coach, a carriage, etc. This question is important because we can perform a similar etymological analysis using our simple 10-word strips of paper. To take a concrete example: suppose now that I have two libraries, and the above process is performed in each, but the only difference is that one library is made up of textual data taken from Wikipedia, which is heavily-weighted toward recently published texts, and the other library is comprised of textual data from Project Gutenberg, which is heavily weighted toward older expired-copyright material. When I constructed my own word-embedding models from the data taken above, I found many cases where analogies diverged between the two data sets. Inputting "Man is to Angry as Woman is to X" in the Wikipedia-model yielded "Angry" whereas the older Gutenberg data yielded the word "Indignant." One can imagine a literary critical analysis beginning with these differences, which would focus on

how emotive states belonged to genders or even normatively prescribed gender via differentials in the potential to cognitively hold certain emotional states.[18]

Our little strips of paper therefore contain a surprising power: the capacity to split data sets along historical periods and recover some form of etymological progression through time. Any reader who has grappled with Heidegger will be familiar with how powerful this form of linguistic archeology can be. And yet, in Heidegger certainly, but even in our small example above that seems to suggest that women could not readily be "angry" until recently, there is a movement from etymology to meaning. We somehow know that the progression of "indignant" to angry *means* something: the word "indignant" contains the implicit reference to "dignity," a concept which seems central to the "novel of manners" of the 19th century. As cognitive humans living in the present, we make this move very quickly, firstly because we rarely encounter "indignant" in everyday language, but also because we know there is something archaic about the notion of "dignity" when is it specifically rooted in a gendered context. And if my possible cognitive states are directly related to the words I have at hand—a concept which does, indeed, seem at the core of my skeleton literary-critical argument above—then, *in theory*, our little strips of paper can tell us more about historical context than we might initially have thought.

[18] Andrew Piper has expressed similar concerns about moving from the "clustering" of concepts that arise from topic modeling to literary-interpretive concepts that invoke some form of "meaning." But he also acknowledges the potential usefulness of differentials between topic modeling clusters across historical periods or authorial groupings: "it is often the case that run well topics capture semantic objects human readers feel comfortable thinking of as topics and that the differential behavior of those topics, whether over time or across different communities/classes of writers, can tell us something about people's behavior." See Piper, Andrew. "All about Topic Modeling." *txtLAB McGill*, January 6, 2020. https://txtlab.org/2020/01/all-about-topic-modeling/.

That caveat—*in theory*—in the above statement is all-important. While the prospect of charting linguistic decay by using various time-sliced datasets is intriguing, that process is technically daunting and there would be many challenges to implementing this sort of analysis on a large scale. But more importantly, I see no way in which that decay could automatically yield any higher-order meaning. We could correlate the decay of clusters of terms over time, as I have done above with the relationship of "indignant" to "dignity." But the move from a cluster of terms to a *concept* is precisely where higher-order contextual cognition comes into play. To put it very simply: *we need to know what we are looking for*. As a cognitive agent, I can make these near-instantaneous moves from indignant to dignity to "novel of manners" to gender simply because I have been trained to look for such things. And this training is rooted, at least in part, to my cognitive assessment of the correctness of gender theory as part of my experience as a being in the world. Now when I invoke "experience" I wish to be very careful, as many theories of cognition implicate, in one form or another, the human body as the mediator of any form of perception. Invoking "experience" in this sense implies "lived experience," the entanglement of human senses with any form of cognitive knowledge. While these conceptions of embodied phenomenology may or may not be true, we do not, yet, need to go that far. I perhaps, almost comically, want to stop our description of cognition as "knowing what to look for." We will explore that concept further, but what we have drawn out in our demonstration of "alien reading" and mountains of paper strips is simply that I have no intrinsic notion of how to fit these strips together to arrive at a concept. If I somehow knew that dignity was related to gender, and so forth, perhaps I could tell you something, but that arrangement exists outside my library. In my imaginary inhabitation of the algorithm, I have no

access to the outside, and all that outside really represents to me is some notion that arranging these strips in a certain way would *mean* something. But I simply do not know what to look for.

Knowing what to look for, then, represents a form of shorthand for a higher-order modality of context that implicates a humanistic orientation toward pure language, an orientation that falls back on learned or embodied knowledge as a method to order phrases into concepts. While word embeddings do express some form of context, it is a context that cannot transcend its fundamental informational unit of the phrase. And yet, considering one facet of the reduced form of merit that I suggest belongs to fan fiction—fidelity to its source material—even this limited form of context can inform an automated analysis of the corpus. It is conceivable that in the *Harry Potter* universe a word embedding derived from the original corpus might yield an analogy such as "Dumbledore is to good as Voldemort is to evil." Given enough of these analogies, one could conceivably construct facets of the *Harry Potter* universe that could be employed in an evaluative capacity to derived works of fan fiction. If the fan fiction text yields a sufficient amount of similar analogical correlations, it is likely that it embodies some notion of the "spirit" of the source texts.

Unfortunately, I found there are computational obstacles here, as word embeddings need a large corpus to accurately infer these correlations. Even the vast amount of textual data contained within the seven original *Harry Potter* novels was insufficient to build the probabilistic space that would yield an accurate word embedding model. This computational

limitation is even more pronounced when taking the much smaller dataset comprised of an individual work of *Harry Potter* fan fiction that we would want to evaluate.[19]

This computational limitation highlights another facet of the relationship of context to merit, one that further underscores the existence of qualitatively different modalities of contextual information. The original linguists who developed the distributional hypothesis theorized that the ambient context of "everyday language" formed the underlying probabilistic space that allow word embedding algorithms to yield information. But novelistic language is very often different from "everyday language," and academically canonized texts often develop this contrast to achieve an aesthetic context. This aesthetic context is something that exists within the much smaller confines of the text itself. My working definition of merit within fan fiction—formulated as a similarity between contexts—thus sits in stark contrast to differential context that often manifests itself within the academic canon. But, crucially, this oppositional context in canonical texts is only apparent if one can organize the differential between everyday language and the language of the meritorious text as an organizing aesthetic structure. In other words, if it were computationally possible to take all of James' corpus and decompose it into a word embedding, and contrast that with the contextual information we can derive from a word embedding composed from the ambient language of his era, we would

[19] Recently, two DH researchers demonstrated that a "deep ML" approach—as opposed to the non-ML based word embedding approach I described here, could indeed model meaningful character relationships in the *Harry Potter* novels based on only the original source texts. As we shall see in the next section, the notion of context in deep-ML based models is profoundly different than that which is used in word embeddings, enabling patterns to emerge within a much more limited set of input data. Vani K, Alessandro Antonucci. "NOVEL2GRAPH: Visual Summaries of Narrative Text Enhanced by Machine Learning." *Text2Story 2019: Second Workshop on Narrative Extraction From Texts*, April 2019. http://ceur-ws.org/Vol-2342/paper4.pdf.

likely find very few illuminating divergences, because the only information we can derive from word embeddings is a result of the repeated juxtapositions of independent phrases.

Now without a doubt, James uses clever and deliberate turns of phrase to achieve a desired effect, as we saw this section's opening example where he begins a sentence with "I was *struck*" as opposed to "It was *striking."* But once again, the units and organizational structure of information here is all-important. Word embeddings cognize phrases but do not consider phrases as *belonging* to anything but an informational whole. Humans must also cognize a similar form of holistic context to develop notions of everyday language, but in interpretive activity, and especially interpretation in the service of critical judgement, must also develop smaller contexts that constitute probabilistic expectations within the text itself. And most importantly, humans must be able to cognize deviations or conformance to these expectations as *meaningful*, a property that emerges only in the synthesis of a phrase as belonging to both the text and to everyday language. In close reading, we therefore situate phrases into a dual contextual arrangement, simultaneously aligning them with the local structure of the text as well as the ambient locutionary conventions of the historical era.

Coming back to our imaginary library of paper strips, we can now picture exactly the width of the gulf between lower and higher order context. Context, in the library of paper strips, is cognized solely as the probability of one word occurring to next to another based on a consideration of phrases as unordered phenomena, organized by nothing beyond the fact that they belong to the same corpus of textual data. But the contextual information derived from phrases in literary-critical activity contains many layers of organization: from the juxtaposition

of phrases to understand the epistemic conceptual arrangements belonging to a historical era, to the local placement of phrases within an individual text in the service of an overall aesthetic.

Our experiment to identify good fan fiction via algorithmic word embeddings has therefore failed on two levels. First, even if I reduce merit in fan fiction as having some fidelity to the source texts, and even though we see how word embeddings can acquire a form of contextuality that could potentially identify "fidelity" as conformance to a set of conceptual and analogical similarity between the derivative and primary works, in practice both the original texts and the individual work of fan fiction do not provide enough information to construct the probabilistic space the supports these computations. Moreover, we have seen how this reduced form of context in fan fiction—based on simple similarity—is very far removed from what is normally associated with the deployment of context within canonically meritorious works. But in this failure, we observe again a qualitative difference between notions of context, and, in our close reading, how context is figured in one class of algorithms, and exactly how much cognitive *excess* is involved in the recognition of higher-order contextual structures in human interpretive activity.

Section 2.5: Recurrent Neural Networks as Cognizers of Local Context

In the last section, we saw how literary merit can be associated with context, and how formulating a reduced form of merit in fan fiction as some measure of similarity between the contextual world of the source texts and their derivatives could potentially be algorithmically quantified. In this experiment, we explored how one class of algorithms—word embeddings—cognize context, which revealed a deep qualitative gulf between lower and higher order

modalities of context. One facet of higher order context that was wholly unrepresented by the algorithm was a form of local context, the notion that words and phrases *belong* to a text.

Taken from a purely informational standpoint, one way of approximating this difference between global and local context is to associate them with different forms of probability. In the global context formed by the probabilistic space of "everyday language" that drives word embeddings, probability is figured as something static: once constructed, the underlying structure driving the word embedding algorithm does not change. The structure does not evolve; there is no way to tell the algorithm that it is wrong or dynamically change its notion of expectation except to rebuild it with a modified corpus. This is perhaps appropriate, as word embeddings ostensibly represent everyday language, a structure that can only emerge from a large and relatively static set of linguistic expectations. But in much smaller context formed by the readerly interaction with a single text, expectations are much more dynamic: as we saw in our first chapter via Ricoeur, the interpretive process is both retroactive and emergent. To put it simply, we change our expectations as we move through a text, and therefore constantly operate under an evolving notion of what is probable. There is a mathematical concept that encompasses some aspects of this evolving expectation given what has come before: *conditional probability*.[20]

[20] Ted Underwood has written about one form of conditional probability in literary character analysis using Bayesian analysis, which is a commonly used statistical technique. See "A Bayesian Mixed Effects Model of Literary Character." *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, 2014. http://www.cs.cmu.edu/~ark/literaryCharacter/. My subsequent analyses of conditional probability will be based on ML techniques to infer the effects of conditional probability as opposed to Bayesian or other statistically grounded methodologies.

In this section, we will explore a particularly complex form of ML algorithms that develop a notion of conditional probability, constantly refining their predictions given what has come before. These algorithms, called Recurrent Neural Networks (RNNs) are not only able to operate with a much smaller dataset, they also recognize the dataset as sequential information, where textual information *belongs* to a local context, a context that is refined in the process of its algorithmic training. Another feature of RNNs that is useful for our analysis here is that they can be configured to output predictions in their training process. For the linguistic-based RNNs that we will employ here, this means that we can configure the algorithm to output passages of text that represent their predictions of what they have learned is probable given the corpora specified as input. We can read this output and gauge its fidelity to the source corpora. This opens a way for us to assess the effectiveness of the RNN's ability to cognize some form of local context, and as we will see later in the section, another method that could potentially identify good fan fiction. In the same spirit as the last chapter, we will "open up" the RNN algorithm and attempt to conceptualize how the algorithm represents this form of local context. As we shall see, this will provide an example of some truly alien reading, one that will hopefully underscore the complexity involved in cognizing the local expectations that evolve as phenomena belonging to a text.

To begin, consider the following passage generated by a trained RNN:

> I said that the sun was a wall of the fire and rose and rose and went out to the room and leaned and sat in the river. They were ran and a dime grinned and should feel the thin whores in the door and he stopped and all but some gentless were watching a slage of stones and went on up a string and stood the barrel to the passing and he stood before it and he looked at her head and reached a drive of white car wooden store settled and with the squire where he was distant over the table.

The first sentence is striking: I like the metaphor of a "sun of fire and rose" and then the switch from adjective to verb in "rose and rose," and the complexity of that structure followed by the blunt simplicity of the conclusion where the narrator "went out to the room and leaned and sat in the river." The next sentence is less coherent but contains some very good imagery: while I cannot quite figure what the narrator is referring to in the phrase "They were ran" and the overall sentence does not quite track, I can tolerate that as a reader for the moment simply because of the middle portion: "thin whores in the door and he stopped and all but some gentless were watching a slage of stones." While I do not wholly comprehend the second sentence, after the terrific opening sentence, the flashes of brilliance in the second sentence, such as the word "gentless," and the effectiveness with which the excerpt as a whole invokes a 19th century American western setting of desert heat, languid despair and latent menace, I feel as if I am in the hands of a competent writer. If was told that this came from a well-known author and forced to guess whom, I'd wager that this excerpt came either from William Burroughs in his late-period, Red Night trilogy or in anything written by Cormac McCarthy.

But no human caused these words to be written: this text was *generated*. Specifically, this text was output by a RNN that I trained on textual data comprised of Cormac McCarthy's early work, the five novels beginning with *The Orchard Keeper* (1965) and ending with *Blood Meridian* (1985). This output, which is one among hundreds of sections I examined and can be readily generated *ad infinitum* now that the RNN has been trained, was stunning to me. Since I knew the source material that was input to the algorithm, I was specifically struck by how these (and many more) generated passages captured McCarthy's style, word choices, and more abstractly, the pervasive mood of struggle, despair and menace. From a stylistic perspective,

the algorithm duplicates some of McCarthy's signature techniques: the lack of punctuation beyond periods, the long, shifting point-of-view sentences conjoined by simple "ands," the archaic-sounding words ("slage," which is an archaic Dutch term meaning "success," but when encountered within McCarthy, one might assume is a geological term derived from "slag") and the enjambments ("gentless").

After recovering from the initial shock at overall effectiveness of the algorithm, I examined the output from a technical perspective and received an equally forceful aftershock. One of the things I admire about McCarthy's writing is his occasional combination of two words into a single, fictitious term that invokes an affective response when unexpectedly encountered within an otherwise familiar locution. That word "gentless" would represent one of McCarthy's greater enjambments. Gentless might nominally be read as a term simply denoting prostitute without a client. But in this particular context, gentless anchors the affective feel of this entire scene: the lack of a client simultaneously suggests desperation (the prostitutes need gents to survive, so gentless is not at all simple loneliness), the utilitarian decadence or degradation of the overall setting (the prostitutes are with or without a gent, a bluntness that infers something mechanical; the suggestion that prostitution is both ubiquitous and passes as the normal mode of intimate relation between individuals) and, finally, the sadness conveyed by the use of the term "gent" as opposed to "client" or "man" (that despite all of this, the prostitutes may in fact harbor the prospect of meeting a "gent" or "gentleman," a humanization that suggests how acutely these individuals might feel about their situation, the heartbreaking notion that they regularly imagine something besides this bleak and mechanical existence).

But the simple presence of the fictitious word "gentless" here was not what disturbed me. What I found inexplicable was that *this term never appears in any of the source texts*. The algorithm, in other words, did not reach into some "bag of words" to get this term. It was *created* by the algorithm itself. The algorithm seemed to not only understand at some level McCarthy's style and his use of enjambments, it created a word *eh nihilo* that fits perfectly within this scene. If McCarthy had used "gentless" next to some description of prostitutes in his source texts, I could understand a way this particular RNN could arrive at this choice. But McCarthy did not use this term. In a classroom setting, I feel I could defend my close reading of this passage and the word "gentless." But close reading often implicitly operates under the (unspeakable) assumption that we are decoding artifacts of authorial intention, moments that demonstrate a great and emblematically human cognitive capacity to produce great writing. It is jarring, then, to close read text that is produced by a machine. But crucially, we must realize that this text was produced by the RNN's probabilistic assessment of what McCarthy *might* write. This could not exist without McCarthy's source works *and* some set of textual features within those source works that align this probabilistic space in such a way that the algorithm produces such coherent and plausible output.

Now our key question in this chapter is whether these qualities in McCarthy's texts that enable the RNN to output such coherent passages corresponds to a form of higher-order cognition of the local contextual space of a text. Or to phrase the question another way, whether the probabilistic expectations developed by the RNN evidence something like the aesthetic expectations that form in the process of human interpretive reading. To answer this

question, we must perform another exercise in alien reading, and explore what it might be like to process textual information like an RNN.

To begin, it is useful to note that the difference between an RNN and a traditional ML algorithm is that an RNN possesses a form of memory. While traditional ML algorithms perform well as pattern matchers, they cannot fully account for the importance of repetition or express the forms of expectation that we normally associate with a local context. RNNs were developed precisely to address this notion of contextual expectation. RNNs account for what has come before as a modifier of what is to come. Like traditional neural network-based ML algorithms, RNNs must be trained and are comprised of aggregations of individual agents. But as part of their training process, each agent must "remember" what has come before and "guess" what is to come next. The correlation of each agent's notion of what has come before as a modifier of what is to come next is evaluated at the end of each round of training. To put it mathematical terms, each agent in an RNN calculates the conditional probability due to prior observations, and the accuracy of that conditional probability as a modifier of the agent's guess is one metric of how relevant that agent is to the system's predictive capacity. Because of the difficulty in representing memory and extending the effects of each agent's memory across many rounds of training, RNNs are incredibly complicated and involve massive amounts of computational resources in their training.

To appreciate this complexity, let us return to the imaginary library we invoked in the previous section, where we explored how word embeddings assembled information from phrases. Our library begins on a vastly smaller scale now, and instead of working with a corpus of thousands or millions of texts, we have only five: McCarthy's first novels. Once again, our

scissor-wielding vandal enters, but instead of cutting these texts into 10-word phrases, he cuts out each *character*, keeping punctuation, spaces and so forth. He lays out these characters on the floor, as an expanse of paper tiles, being careful to maintain the sequence of character-tiles as they occurred within the text. Once finished and arranged sequentially, however, he collates all these paper tiles in one conjoined mass, so while the characters are still ordered correctly, we have no idea where one text begins and the other ends. Satisfied, he leaves and ironically wishes us good luck.

At an average of 3000 characters per page, we have roughly *six million* paper tiles lying on the ground. So just taken as aggregate pieces of data, our tiny corpus of five novels has been cut up in such a way to produce a similar number of informational elements that existed as novels (or even phrases) within our enormous library of word embeddings. But each element contains much less information, a character instead of a 10-word phrase. In our other library, buried in what now seems like enormously dense strips of information, we could arrange these strips in such a way to figure out how words were normally used within the context of meaningful phrases. But we can recover none of that context here to read these character tiles. To coax some form of information out of this arrangement, then, we will need to perform some very strange phenomenological contortions.

The first thing we do is create what we will call agents, constructing one agent for each character in my alphabet. These agents are simple things and we tell each one what to look for. We create one agent to looking for a capital 'A', one that is looking for lower case 'b', one that is looking for a period, and so forth. Let us suppose that this results in several hundred agents, given that we want to look for letters, numbers, punctuation and so forth. These agents don't

have any awareness of one another, but we decide that we need some oversight, so I create a very special agent that can sit and oversee these hundreds of autonomous searchers. Picture this special agent perched on the tallest vantage of the library, as overseer, conductor and composer. In honor of our source material, let's call this special version the Judge. The Judge gives each agent identical copies of the millions of character tiles in the library, and then issues a bizarre proclamation. Each agent must now make *additional* copies of themselves, but he wants these copies to be aware of one another. Specifically, the Judge wants these sub-copies to stand on one another's shoulders, as many as possible. The ceiling of the library limits this number to twenty, and this multitude packs the library in a bizarre but organized formation: hundreds of agents each standing at the beginning of millions of tiles, each of which is supporting a stack of nineteen duplicate agents on their shoulders. The Judge issues a stern warning: the stacks of agents can communicate within their vertical stacks, but cannot communicate with adjacent stacks in any way, shape or form.

Thus organized, the Judge hands every agent a bag that can hold the last 20 characters it has encountered. Now equipped and in formation, each of the hundreds of 20-high stacks of agents are told to march forward one step in the sequence of character tiles. At each step forward, the bottom agent can copy the tile on the ground in front of them and put it in their bag. Each bag only holds 20 character-tiles, however, so at each step, the agent at the bottom must remove the oldest tile and replace it with the one just encountered. Now at each step, instead of just throwing that character-tile away, the agent at the very bottom hands his oldest tile to the agent standing on their shoulders. The agent at the top of the stack must eventually discard the tile because there is no one to hand it to.

At each step, the Judge asks only this of each stack of agents: tell me how likely it is that you will find your character *given* what you have in your bag? After the first step, only the agent at the very bottom can reply, as it is the only one with any information in its bag. But after the first step, that bottom agent has nothing to base *anything* on, and cannot really offer a coherent opinion as to what might be next. It's the first step in this bizarre library, after all, and none of these agents, and not even the Judge himself, has any notion of grammar, syntax, spelling, words, or anything resembling higher-order linguistic structures. After this first step, then, if I am that unfortunate agent at the bottom of the stack, and I have no remote idea what the characters on these tiles represent, I have no real information at all, except that I'm looking for the character 'b' and I just saw the character 'K'. But the Judge demands an answer, so without knowing anything at all, even how many copies of similar stacks might be marching alongside me, I might reasonably state that "I think the next character has a 0% likelihood of being a 'b'." I have never seen a 'b' before, why should I expect to ever see one?

Now the Judge orders a second step. If I'm the bottom agent, I duplicate and hand that initial 'K' I saw in the first step to the copy on my shoulders, step forward, and collect a 'b'. Now as the agent at the bottom of the stack, my extremely limited worldview is modified dramatically. I have a 'b' and a 'K' in my bag now, so given this, and only this, I might think there are only two letters in our universe, and I can now say with more confidence that there is a 50% chance my next step might reveal a 'b'. But the Judge is stern, and doesn't consider my individual opinion, he only cares about the opinion that represents the collective opinion of my entire stack. The agent on standing on my shoulders only has that initial 'K' and thus offers the same opinion I initially did. No other agent else has any information yet, so they are not able

to answer, but collectively we now have two individual opinions in our stack, and they must be combined into a probability.

This process marches relentlessly on and on through the end of the list of tiles, and the Judge is keeping score. Considering the collective opinion of each of the hundreds of stacks, he contains information about each step these stacks have taken as a collective whole. After each step, he is the one that chooses which stack is most likely to be correct, and given this, he issues his pronouncement as a single character, the one most probable to come next. "Of this is the judge judge and the night does not end"—and at least for these agents that is true, for he forces them to march and march the sequence of tiles again and again, millions of tiles traversed millions of times by hundreds of agents stacked twenty high.

Let us recall our opening quote generated as the output of this process and try to imagine how that could possibly come out of this library. There exists no notion here of anything resembling rules of grammar or syntax, no vocabulary of words nor anything near a notion of what might constitute a common phrase. There is true alterity here, as it is nearly impossible to conceive how this process, carried out at the character level, could ever result in the opening quote. There are, admittedly, grammatical mistakes in that quote, and we can now see why those exist, as we have nothing in the way of grammar here except what might arise from this stepwise process. But for this process to create *new* words, and furthermore a new word as appropriate as "gentless" *feels* in that first quote, is deeply mysterious.

The conceptual dissonance here is likely related to the complete lack of anything resembling human reading in the RNN. The word embedding library and its strips of paper

could indeed be analogized and extrapolated into commonsense notions of reading, but it I can think of no ready analogy here. This fundamental alterity is pointing us toward something odd about cognition. The core concept at work within an RNN is *expectation* manifested as conditional probability. I have suggested that expectation might be reducible to some notion of local context. An RNN is based on an extreme form of local context, the context formed by the proximity of characters next to one another, expressed as a function of the cluster of characters that have come before. But given that we are limited in the number of characters we can immediately access as direct context—recall that our imaginary agents' bags could hold only 20 characters—it is difficult to align what an RNN considers context with what we, as readers, would consider context.

Recall how this expectation is formed in our exercise in alien reading. Every agent in the stack is passing its last observation upward, so each has a slightly different notion of context, each moving one step behind the other. There is therefore a residual expectation contained within each stack, as the agent at the top of the stack is expressing an expectation given a different (and older) set of expectations than the agent at the bottom. This residual is furthermore accretive as the Judge gathers the stacks' collective expectations together into the decision that precedes his pronouncement of which character is to come next.

This residual in RNNs builds over millions of iterations over the same set of characters. It gathers its predictive power as the result of a process so computationally intensive as to be inconceivable. But I suggest that what I have alternately termed as local context or expectation, as we experience it as cognitive readers, must contain the same accretive elements. I suggest this precisely because the fundamental inexplicability of RNNs. We know

that they eventually do produce output like the opening text, and we know what they are only working with individual characters, so we can conclude that an RNN is effective *only because* the entire system contains a residual element—a collective memory—that grows over eons of computational time.

Returning to the questions of merit and context, RNNs, despite their complexity, cognize textual information in similar ways to the word embedding algorithms explored in the last section. But whereas word embeddings use huge collections of phrases to predict the likelihood of words, RNNs use a much more involved process to predict probable groupings of characters within a much smaller dataset. In both word embeddings and RNNs, then, context manifests itself as an underlying structure of probability. Yet RNNs seem to be able to do so much *more* with this probability, bootstrapping everything from words, grammar and sentence structure to much higher-level features such as authorial style from the most basic textual informational unit imaginable: the single character. The ability of RNNs to build such complicated structures points toward the powerful role memory occupies in the cognition of context. To propose a deliberately provocative analogy, the ability of an RNN's memory to modify its probabilistic expectations as it encounters new data can be compared to a very restricted manifestation of phenomenological time that was central in our analysis in chapter one. An RNN constructs a narrative of characters, a "story" that, however meaningless to us, nonetheless represents some form of cognition of context as an emergent property, a retroactive assessment of prior observations as a potential *cause* of the informational whole.

This suggestion that RNNs cognize a form of phenomenological time invites us to consider what RNNs cannot reproduce as a measure of the limitations of that cognition. As we

have seen, RNNs can output remarkable passages of text, but they are manifestly unable to organize these passages together into anything resembling a coherent narrative. This limitation points precisely to the problem within the dubious analogy above. If contextuality is defined as something wholly probabilistic—even as the highly complex conditional probability that emerges within RNNs—we omit the fact that humans cognize and interpret narrative structures in texts as not as what is *likely*, but rather as what is *sensible*. While human readers do reference context as a set of likely societal, behavioral and textual expectations, the sensibility of context as an *aesthetic* phenomenon often arises from a consideration of what is *unlikely* within a text. This higher order modality of context must not only define a set of expectations across a wide set of heterogeneous information—historical and societal norms, embodied intuitions, narrative conventions, and so forth—but also organize it as an oppositional structure, a backdrop that allows unexpected things to emerge as aesthetically sensible phenomena.

So while RNNs seem to capture many facets of lower order context in their ability to reproduce limited amounts of text that possess an uncanny fidelity to their source material, the contextuality required to organize these small units into an aesthetic whole belongs to a much higher order form of contextuality. Yet in the ability for RNNs to capture something so abstract as authorial style, we can see that at least some artefacts of traditional literary critical study manifest themselves as simple probabilistic structures, expectations that can be wholly modeled by—and therefore are contained within—the textual data itself.

The fact that we normally assess style as *belonging* to an author opens a way that we could potentially utilize RNNs to assess some form of merit, especially within fan fiction. If an

RNN is trained on the set of original *Harry Potter* texts, and another RNN is trained on a one or more derivative works of fan fiction, then comparing the two sets of RNN output would give some subjective measure of how well the fan fiction author has captured the "spirit" of the original works. There are computational obstacles here—RNNs require an exceptionally intensive and prolonged training period—and, more importantly, we cannot factor out the human interlocutor required to make the interpretive assessment that judges the fan fiction output against the source output. But we have massively reduced the amount of *human* effort required in this assessment, as one can reliably judge merit based on the condensed representations output by the RNNs. And furthermore, insofar as certain professional genres of writing—legal discourse immediately comes to mind—embody a set of conventional stylistic traits, one could conceivably extend this semi-automated assessment into other fields and applications beyond what we have discussed here.

As always, one must be careful with harnessing technology to evaluate artifacts of human production, and while the first approximation of textual fidelity that could be read into sets of RNN output can be useful *as a summary*, the fair evaluation of textual merit must still involve an interpretive engagement with the text as a whole. This final thought underscores the modalities of higher and lower order context. Lower order context can yield expectations as they manifest themselves within wholly textual information. There is another, deeper form of context that is comprised of the differential between expectation of surprise, a structure that emerges only as an appreciation of a text as a modifier of its context. The difference between context-as-probability and context-as-aesthetic is another way of describing the qualitative gulf between lower and higher order cognition of context.

Section 2.6: Conclusion

In this chapter, we have taken one of the most abstract literary critical terms—context—and explored how algorithms can model contextual information in the service of the critical judgment of textual merit. Context is fundamentally problematic as it expresses some relationship of a text with the external world, and it is difficult to figure even how we, as humans, have access to non-experiential contextual idealizations such as a "historical milieu" or a set of "societal norms." By applying algorithmic assessments of context in the service of answering a question—what should I read in a huge corpus of unexplored texts—we have seen three ways in which machines can construct utilitarian models of contextuality: 1) as the collective opinions of other readers determined to be similar to me, 2) as an ambient backdrop of everyday language that can confer a form of meaning to a word by analyzing its placement within phrases and, 3) as the local manifestation of conditional probability given what has come before in the local space of a text. By thinking of what it would mean to read like each of these classes of algorithms, we have observed not only the shortcomings within each computational assessment of context, but also how difficult it is to model context as a purely informational trait derived from textual data alone. As we saw in the first chapter's analysis of sequence, our analysis of context here points to qualitatively different modalities of higher and lower order context. Reading algorithms—understanding algorithms not solely as tools but as examples of different, and perhaps fundamentally alien, cognitive engagements with texts—is a way for us to explore and appreciate the complexity of context as it is deployed in humanistic literary interpretation.

Chapter 3: Reading Algorithmic Text and Paying Attention

Section 3.1: RNNs as a Collaborative Space Between Humanists and Technologists

In chapter 2, we worked through an extended demonstration of how the identification of literary merit in a massive fan fiction corpus could be productively framed within the contours of DH without yielding to the conventional argumentative arc of dataset/statistical result/model/interpretation. In this alternate vision of the methodological possibilities of DH, my goal was to show how knowledge about the *tools* of the trade—the code and algorithms that generate result sets—could open equally fruitful avenues into the humanistic problems at hand as the result sets themselves. Underpinning this argument is the conviction that the algorithms used in DH analyses have reached a level of sophistication that allows them to be conceptually inhabited: i.e., we can productively imagine what it might be like to "read" like a machine. This imagined phenomenology—the Nagelian question of "What is it Like to Be an Algorithm?"—can be used to productively reinterpret longstanding problems within the humanities.

In this chapter, we will maintain our focus on RNNs—one of the most complicated species of ML algorithms in-use today—to explore how this conceptual inhabitation can open collaborative spaces where humanists and technologists can collectively advance ML research. As we have seen, RNNs can output text after they are trained on a source corpus. When an RNN is trained on texts written by a single author, literary critics can engage with the algorithmic output on a humanistic level, readily identifying textual features and authorial idiosyncrasies that are cognized or missed by the machine. This humanistic knowledge can be

bought into contact with the technologists' understanding of how the algorithm is coded and configured.

The next chapter will walk through several experiments that demonstrate how this collaborative space could operate. Many RNNs can be readily configured by adjusting relatively high-level code and parameters, and these adjustments can lead to qualitatively better textual output. Understanding what these algorithmic "knobs" are tuning from a technical perspective and combining that with a humanistic or aesthetic appreciation of the differences within the ouput will lead us once again into a contemplation of different orders of cognition. As we conduct our experiments, I will attempt to approach what we have seen as the characteristic quality within RNNs—memory—to notions of *attention*. Attention within ML models has been the subject of much research within the technical community, as part of what seems to differentiate human memory from RNN memory is the former's ability to *selectively* retain events that are cognized as important.[1] From a technical standpoint, our experiments in the next section will show us how vital the extension of memory is to an RNN algorithm and give us an idea of the improvements that could be made if an RNN could be configured to only retain important information. In the concluding section, we will consider a humanistic assessment of how attending to certain facets of texts becomes vital to the organizing process of interpretation.

[1] For a technical overview of RNN attention, as well as a proposed implementation, see Vaswani, Ashish, Noam, Parmar, Niki, Jakob, Jones, et al. "Attention Is All You Need." arXiv.org, December 6, 2017. https://arxiv.org/abs/1706.03762

Section 3.2: Tuning, Reading and Evaluating RNNs

In this section, we will explore the contours of the "unreasonable effectiveness"[2] of RNNs by example. In the last chapter, I described the results of trained RNNs as "shocking," a sentiment shared by others in the technical community.[3] I cannot fully comprehend how RNNs move from a set of characters to fully formed passages *as the result of an algorithmic process*. To understand the unreasonable effectiveness of RNNs, then, we must gauge their effectiveness from a vantage that is, at least in part, humanistic. As we will see, RNNs fail to capture some authors effectively. If trained literary interpreters can locate authorial or formal idiosyncrasies that are consistently eluding the RNN, then they can provide an avenue for technologists to understand why these algorithmic black boxes are failing.

To this end, let us begin by revisiting two felicitous passages generated by trained RNNs.

Generated Shakespeare:

> KING HENRY VI:
>
> Sir, there is no thought out of semblance, the man are the first of the state, which the dignity of my house.
>
> Most sea, and slandering the mind of beard, and has won the stars,
>
> The contention we do grace the little noble man
>
> In the content, and the offence of my command?

[2] A phrase coined by one of the leading contributors to RNN technology in a widely influential blog post. See Karpathy, Andrej. "The Unreasonable Effectiveness of Recurrent Neural Networks." May 21, 2015, http://karpathy.github.io/2015/05/21/rnn-effectiveness/.

[3] Karpathy: "Sometimes the ratio of how simple your model is to the quality of the results you get out of it blows past your expectations, and this was one of those times. What made this result so shocking at the time was that the common wisdom was that RNNs were supposed to be difficult to train (with more experience I've in fact reached the opposite conclusion). Fast forward about a year: I'm training RNNs all the time and I've witnessed their power and robustness many times, and yet their magical outputs still find ways of amusing me." (ibid.)

MARCUS ANDRONICUS:

My lord, my fault was fair,

And crowns to defend the sound

Generated Cormac McCarthy:

> I said that the sun was a wall of the fire and rose and rose and went out to the room and leaned and sat in the river. They were ran and a dime grinned and should feel the thin whores in the door and he stopped and all but some gentless were watching a slage of stones and went on up a string and stood the barrel to the passing and he stood before it and he looked at her head and reached a drive of white car wooden store settled and with the squire where he was distant over the table.

With these passages in mind, recall our thought experiment where we imagined an RNN as a process taking place in a cavernous library, with an expanse of paper tiles covering the floor, each tile inscribed with a single character taken from the source texts. We configure an array of agents aligned horizontally, each agent facing their own copy of this massive expanse of identical paper tiles; each looking for a specific character. We then duplicate the agents *vertically*, asking them to stand on one another's shoulders. We hand a bag to every agent that can contain a limited number of tiles. We grant the agents the ability to copy the character-tiles and place them in their bags, but with the provision that they must copy the oldest tile in their bag and hand it to the agent sitting on their shoulders. If their bag is full, the agent in the vertical stack must discard the oldest tile in their possession. Finally, we perched a very special entity at the top of the library—the Judge. The Judge asks each stack of agents to march forward through the expanse of tiles, and at each step asks each stack their opinion of the likelihood of their character coming next. The stacks of agents can discuss this opinion vertically and collectively come to an opinion, but it is the Judge who decides which stack is

correct. At each step the Judge examines his own history and issues a proclamation of what character is to come next. Then and only then can we look up and see the next character tile and determine whether the judgement was correct. If the Judge guessed the correct character, he proportionately trusts the stack when he feels similar circumstances prevail; if he was wrong, he learns from his mistake.

The imaginary experiment above captures the spirit of two important characteristics about RNNs: 1) that they are *generative*, meaning that they output something as they are trained (here, it is the Judge's proclamation of what character is to come next) and, 2) they possess multiple layers of memory—the narrowly focused, short-term memory that held with the bags of each agent within a vertical stack *and* the more occult, longer-term memory that accrues from the Judge's account of which stack is correct given a larger view of the current state that prevails within the library of agents and tiles.

Considering this metaphorical example, we can readily explore several different ways in which real-world RNN algorithms can be configured. In the following subsections, we will examine three common "hyperparameters" used to configure these algorithms and provide examples of how these different parameterizations affect the output.

RNN Hyperparameter Case #1 : Character vs. Word

Linguistic-based RNNs can be trained at the *character* or the *word* level. The above examples were generated with a character-based RNN. Recall that in our metaphorical library, we cut each character out of the source texts and placed them on the floor as individual units of information. But we could just as easily cut out each individual word (as we did in the

imaginary topic-modeling library). It is not clear which is preferable from a technical standpoint. On the one hand, if we work at the character level, we can dissect each text into a greater overall quantity of informational units and constrain the range of the units to the range of characters (a set that, in the English language at least, is normally bound by the range of 256 ASCII characters). On the other, if we consider each individual unique word in our corpus, we have a much larger range of possibilities (on the order of, say, 50 thousand unique words) but we are, of course, working with a much denser unit of information. Consider the following passage, generated from the same set of Cormac McCarthy novels as the character-based example above, but trained at the word-level:

Generated Cormac McCarthy (trained by word):

> The kid gestured and spat. Glanton pushed through the officers and wore alone up into the flanks. When Glanton rode out among the pandemonium of the first clay small strobes of white pushed around everywhere in the ceiling and the rifleball five, six of them. They crossed and he woke with the dried sun hung pushed toward the sand and showed the cylinder and set it astride one and they did not look. Perhaps he'd heard the indian among his fingers, passing through the ground by the city.

With a literary critical eye, compare the first character-trained passage with the second word-trained passage. In the word-trained output, we see the emergence of many more individuals from the source texts (e.g., Glanton, the kid) as well as a much more familiar plotline: the quintessentially anti-authoritarian Glanton bringing about "pandemonium" by "push[ing] through the officers" and firing his rifle indoors. In the character-trained output, we can still see the emergence of typically McCarthy-esque themes: an affective feel of overall

desolation and despair personified by the “thin whores in the door” as well as words that do not exist any of the source texts (e.g., “gentless”) but nevertheless seem to fit McCarthy’s style.

Which is better, the word-trained output or the character-trained output? This is, of course, a loaded question. If you phrase this question technically, it can be quantified exactly: one RNN is better trained than another when it recapitulates the source texts with more fidelity. A perfectly trained RNN on *Blood Meridian* (1985) would output *Blood Meridian* exactly, sequentially, word for word and (by extension) character by character. But aesthetically, the value of these RNN passages lie precisely in their failure to reproduce the source texts. As a reader, I can identify with the word-trained output more readily, as it contains proper names and generally references the source texts in a more concrete fashion. But I appreciate the character-trained output much more, as it seems to capture a certain essential style, a setting, a mood—familiar, but not wholly derivative of any known work by McCarthy.

This personal aesthetic assessment is pointing us toward a technical facet within the algorithm itself. RNNs were developed precisely to overlay a form of memory on top of traditional ML pattern matching. In our metaphor of the character-tile agents and the Judge, this memory resides both in the bags of each agent and the residual accretion of the Judge’s higher-order assessment of each stack’s validity, an assessment that is continually refined after each step within the library. There are two levels of memory at work here, and indeed the variant of RNN that I used to generate these passages uses a technique involving LSTM (Long Short-Term Memory). All RNNs are “feedforward” networks, meaning that the past observations of each cell (in our metaphor, the bag that each agent carries) is fed into the next

step of the algorithm, which forms the "recurrent" informational state within the RNN. LSTMs add an additional layer to this network, which utilizes a technique called "backpropagation through time" (BPTT). BPTT expressed mathematically is fairly involved, but in terms of our metaphor, we can understand it as each agent within a vertical stack assessing the error within previous steps. Recall that at each step in the library of character tiles, the bottom agent copies the character just encountered, places the copy in its bag and hands the oldest character to the agent above. The agent at the very bottom of the stack is therefore encountering newer information than the agent sitting on top of the stack. BPTT allows the agents to look upward, in effect, accounting for the errors of agents with older information before calculating their own predictions. Feedforward with BPTT thus makes a sort of Janus out of each stack of agents, enhancing their collective predictive power with the benefit of hindsight. This algorithmic twist allows each agent to "remember" much more than what they carry in their bag by more than an order of magnitude. Each agent may carry a bag that holds only 30 characters, but with BPTT, their predictive assessment can span a thousand or more observations.

Now place this concept of extended, bidirectional memory into context with the two McCarthy passages generated at the character and the word level. The key thing to note is that the overall algorithmic process has not changed, but the *unit* of information has. In the context of our metaphor, when trained at the word level, our stacks agents begin facing a similar expanse of tiles, but instead of characters, these tiles are inscribed with full words. Because there are many more discrete *words* in our corpus of McCarthy's works than discrete *characters*, we must create many more stacks of agents, perhaps tens of thousands that look

for each unique word used in the source texts as opposed to the two hundred or so looking for individual characters. Each stack is the same height, but a library that holds word-agents must be expanded horizontally more than a hundred-fold to accommodate the additional rows of agents. From a computational perspective, we need to allocate much more space (i.e., memory) to the overall process to accommodate the increase in discrete informational units.

In addition to this computational difference, there is a far subtler and more important difference between RNNs trained at the character level versus the word level. This is best illustrated by example. Suppose I am training a character-based RNN on literary texts. I create a stack of agents looking for the open parenthesis character: '('. If this vertical stack of agents is properly trained and it marches forward and finds the '(' character, it should *not* expect to find the '(' character again until it has encountered *enough* non-'(' characters. "Enough" should come to mean approximately the average length of a parenthetical clause within the given source texts. David Foster Wallace aside, most writers do not nest parenthetical clauses. The construction "the (happy (but lazy)) dog" is simply unwieldly. Now if this RNN were being trained against textual data taken from mathematical texts or computer code, it would be far more likely to encounter an expression such as "x + (4 * (y – 2))." One critical thing to note here is that the vertical stack of agents is only expressing its collective expectation of the '(' character in its prediction. Another stack is predicting the likelihood of the close parenthesis character: ')'. It is the Judge who oversees the stacks as a collective and issues the proclamation of what character is most likely to come next. If the system as a whole has just encountered a '(' character, then the Judge will come to observe that the stack predicting the ')'

character will begin to indicate a stronger prediction as it marches forward toward the average number of characters within a parenthetical clause.

There are two corollaries to this example. The first is that the extension of memory within the vertical stacks is critical. If my vertical stack of agents is thirty high and each holds a bag of thirty characters, but the average length of a parenthetical clause in the source texts is sixty characters, then my vertical stack can only issue an accurate prediction if it can extend its outlook past its immediate duration. In computational terms, correctly configuring the amount of direct memory and algorithmically encoding enhancements such as BPTT are perhaps the most important choices in RNN design. The second corollary is that I can only predict things that occur within the source texts *and* which occur within the extended horizon of the stack's memory. If the character ')' never occurs in the source texts, the system will approach a zero likelihood of it ever occurring.[4] But more subtly, if collections of characters—in our example, the '(' and ')' of a parenthetical clause—only co-occur in strings of greater length than the extended memory of the vertical stacks, then the RNN can never learn to emit a properly formed parenthetical clause.

With all of this in hand, we can now conjoin a computational perspective to our aesthetic assessment of the above passages. In the passage emitted from the character-trained RNN, the word "gentless" is used to describe the "thin whores in the door," a locution I found to be entirely appropriate to McCarthy's style. As I have mentioned, the portmanteau "gentless" never occurs in the source texts, and we can now understand why it is possible for

[4] In fact, in Cormac McCarthy's early works (up to and including *Blood Meridian*) parentheses only occur in his very first novel, *The Orchard Keeper* (1966).

the character-based RNN can emit this word, but not the word-based RNN. From the opposite direction, we can also now see why the word-based RNN emits passages the contain more familiar characters and plotlines. "Glanton" and "the kid" often occur next to one another, considered both as words and in the fictional structure of *Blood Meridian*. Both the kid and Glanton have often either "gestured" or "spat." And although no one has ever "gestured and spat" in the source texts, it is understandable why a word-based RNN might calculate that it could be probable that "The kid gestured and spat" as the algorithm has no notion of the likelihood of a *phrase*. We are working entirely within the realm of probability here, and as we increase the density of the informational unit, we make a tradeoff that can be understood both computationally and aesthetically. It is precisely this conjunction that interests me here. When I say that I enjoy the output of the character-based RNN more than the word-based RNN, what I specifically like is the unexpected familiarity of the passage. The *image* of the "gentless" "thin whores in the door" belongs to McCarthy's writing, but the *sound* does not—one can readily imagine this grim setting in his work, but not the playful rhyme. What is aesthetically pleasing to me is probable using one informational unit (the character) but impossible using another (the word).

In this example of choosing the hyperparameter to configure the RNN to learn by character versus word I consciously experience different affective responses from the output of the same core algorithmic process. The core quality here seems to be an aesthetic of *expectation*. Readers familiar with McCarthy's works may share my appreciation of the character-based RNN output, as it presumes an appreciation of the stylistic and thematic qualities of the source works. Readers unfamiliar with McCarthy may find the word-based

passage to be more pleasing, as it contains higher-level plot and character structures that frame the action in a manner that many would find appropriate to the genre.

RNN Hyperparameter Case #2: Number of Epochs

Since character-based RNNs seem to take us into more complicated areas of aesthetic or contextual expectation than word-based RNNs, we will hereafter confine ourselves to the output of character-based training. As one can imagine, it is computationally intensive to train an algorithm working at the character level into a system that must acquire some form of information about grammar, spelling, syntax and higher-order stylistic structures. In our metaphorical library, this information is acquired by forcing the stacks of agents to march the sequence of tiles again and again. Since the RNN algorithm uses a bidirectional learning system that combines feedforward acquisition with backpropagation through time, it can gain information traversing the set of tiles both forward and backward. A traversal of the entire array of agents back and forth through the entire sequence of character tiles is called an *epoch*.

Fortunately, because each stack of vertical agents only communicates within its stack (i.e., they never care about the stacks looking for different characters marching horizontally alongside of them) this training can be done in parallel. We can, in other words, allow each stack of agents to march alongside each other and collate each of their results at the end of the calculation. Yet even in a character-based RNN, there are usually hundreds of stacks marching in formation, one for each distinct character in the dataset. This computation is prohibitively expensive on standard CPU-based hardware, which can only run a single calculation per CPU core (a very performant modern desktop computer can have eight cores, a ten thousand-dollar

server can have 32). Perhaps surprisingly, a solution designed for modern computer-based gaming has solved this computational issue for us. Dedicated high-end graphics cards (GPUs) contain many thousands of cores, which, in graphics-intensive gameplay, carry out the vast amount of calculations necessary to render complex scenes at fast framerates. The limitation of GPU cores versus CPU cores is that the GPU cores are normally constrained to very frequent, atomic calculations that require relatively little stateful memory and do not communicate with one another. As it turns out, the calculations involved in most ML training fit these constraints perfectly, making GPUs the standard hardware choice for most forms of "deep" ML.

RNN training is one of the most demanding tasks in the ML ecosystem, as each epoch of training involves billions of individual calculations. GPU programming was formerly a dark art, requiring low-level knowledge of the hardware and programming language extensions.[5] Fortunately, again, with the rise in popularity and commercial viability of ML, several toolkits have appeared that ease these programming tasks a great deal. The character-based RNN used in these examples was built upon one of these libraries—Google's *TensorFlow* toolkit—which exposes several training parameters. One can not only configure the number of epochs used to train the dataset, but also insert code to output samples as the RNN traverses each epoch. For example, training the RNN on the collected works of Jane Austen, we have, at epoch 1:

> Yele the wherd feryurind an the ast he hive cham nit an ho
> redingand. Anle blengidgange tile so mes isd berse corsissary whr ceret titt il
> srey f sile tome as ame oh meny fer hachintoure o thenderasisirat.

[5] GPU programming was used heavily in the financial space and, more recently, in Bitcoin mining. Both industries have moved on, however, to specific chip-level hardware tailored exclusively for their calculations.

At epoch 10:

> as they will be the most agreeable and impossible. They had not been
> able to return to Miss Bitgley, who had not been absent, and that he
> had been at home, and her sister was always the first to his father and her father

And finally, stabilizing around epoch 50:

> When her father's acquiescence had been the case, and want the convenience of their consequence, that she had never been so different as he could. She had never seen that she had not the small of their being sensible of the influence of her friends and the play in the course of his own.

After viewing many of these progressions, what I find remarkable is that authorial *style* seems to emerge relatively soon in the training process. In epoch 1, we can see exactly how little we start out with: we have no spelling or grammar and the passage is gibberish. And yet, even here, we can see a certain cadence to the characters. The length of words and the spacing seem to replicate normal sentence patters – we don't see, for example two spaces between words and the periods occur about where one might expect. In the tenth epoch, we still have relatively simple sentence structures, but we have one firm instance of an Austenesque locution—"agreeable and impossible"—and another more curious phrase that is logically tangled yet stylistically somewhat familiar: "her sister was always the first to his father and her father." Finally, in the later epochs, we see much more familiar prose, with entire sentences

falling both grammatically and stylistically in-line: in particular, “want the convenience of their consequence,” a very Austenesque phrase that never occurs in Austen.

RNN Hyperparameter Case #3: Sequence Length

While thinking about sound and rhythm of language I happened to re-view Stanley Kubrick’s cinematic adaptation of *Barry Lyndon* (1975). I was struck by the film’s ability to effectively capture Thackeray’s dry wit through the juxtaposition of the deadpan narrative voiceover against the lush soundtrack and extravagant visual set design. Based on this connection, I reread several of Thackeray’s novels and decided to run them through the RNN.

Up to this point, in Shakespeare, McCarthy, and Austen, we have seen examples of well-behaved RNN output that captures important facets of the source texts. But after reviewing the output of around a hundred authors, I would say that approximately half were not captured well by the RNN. Thackeray is one of those authors that resisted recapitulation by the RNN, alongside others such as Henry James, Pynchon and Hunter S. Thompson.[6]

<u>Generated Thackeray-30:</u>

> The woman was a good deal, says Giglio, and that when I was a good deal of me that I was not a widow the same men who went to see me as a good soul, I shall be a good deal of men of my familiarity and a good deal.

[6] Hunter S. Thompson is an interesting case. It is well known that he typed out Fitzgerald’s *The Great Gatsby* (1925) “just to get the feeling of…what it was to write that way.” Menand, Louis, and Louis Menand. "Believer." *The New Yorker*. June 20, 2017. I am a strong proponent that there is a connection between the tactile interplay of fingers/keyboard and style. After encountering this quote, I was struck by the similarities in style between Fitzgerald and Thompson. It was surprising, then, to see the RNN perform well with Fitzgerald but not with Thompson, as I consider their writing to be stylistically similar. I think the question of genre is important here: Thompson, whose work as a journalist outweighs his prose output, is largely working in longer sentences that are not punctuated by dialogue (as Fitzgerald’s novels are) or stanza-like sequences (Shakespeare). As we shall see here in this section, because of typical length of first-person discursive structures, it is difficult to for character-based RNNs to maintain enough memory to effectively train themselves across longer, unbroken spans of text.

Generated Thackeray-60:

> Prince Guldo, and his father's conversation with a smile, and to the profession of the street of the country which he was to be seen in his face, which he would have been too morally attendant, and had been a man of the world to the poor old fellow, as the latter was the second profession.

Neither of these passages make much sense, although I would claim that the second seems more coherent. The difference between the RNN that output the first passage and the second is that the former was configured with the hyperparameter "sequence length" set to 30 and the latter set to 60. In our metaphor, sequence length is the height of each vertical stack of agents and the capacity of their bags of characters. Technically, this parameter expands the memory of each node so that characters further away sequentially within the texts can influence the probability of the estimate at each step through the training sequence. Going back to our first subsection, recall that with a sequence length of 30, when one accounts for the extensions provided by BPTT, agents can consider patterns that occur in sequences of approximately 1,000 characters. As we saw, if parenthetical clauses occur within spans of that length within the source text, the system will learn that a proper literary parenthetical clause begins with the '(' character and ends, after a sufficient amount of intervening characters, with a ')' character. Doubling that hyperparameter from 30 to 60 roughly doubles the sequence length that can affect each node's calculation, and we can metaphorically understand this as a form of extending the duration of each individual agent's "attention span."

Thackeray's writing is quite discursive, even taken within the relatively digressive style prevalent among his Victorian contemporaries. Take, for example, this passage from *The Luck of Barry Lyndon*:

> About this time, it must be premised, the United Kingdom was in a state of great excitement from the threat generally credited of a French invasion. The Pretender was said to be in high favour at Versailles, a descent upon Ireland was especially looked to, and the noblemen and people of condition in that and all other parts of the kingdom showed their loyalty by raising regiments of horse and foot to resist the invaders.[7]

And contrast it with the famous opening passage from Dickens' *Tale of Two Cities* that describes the same historical moment:

> It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, it was the epoch of belief, it was the epoch of incredulity, it was the season of Light, it was the season of Darkness, it was the spring of hope, it was the winter of despair, we had everything before us, we had nothing before us, we were all going direct to Heaven, we were all going direct the other way…[8]

Finally, consider the following output from an RNN trained on Dickens' works (with a sequence length of 30):

<u>Generated Dickens-30:</u>

> You may speak to be satisfaction to touch the best of that country, and that I have been a little man to be surprised, and that little fail was a service to the children. It was not to be so much of the conversation of the chair of the shop, the shop had been the share,

[7] Thackeray, William Makepeace. *The Luck of Barry Lyndon*. Edited by Walter Jerrold, 1844. Project Gutenberg, www.gutenberg.org/files/4558/4558-0.txt.
[8] Dickens, Charles. *A Tale of Two Cities*. Edited by Judith Boss, 1859. Project Gutenberg, www.gutenberg.org/files/98/98-0.txt

> and there was not a share in course of his face. He said to the first thing that I should be a good friend, that I was not and that I had been.[9]

While it is not quite clear what the generated Dickens' passage is referencing, it tracks well and contains some aesthetically pleasing symmetrical constructions: "that I should be a good friend, that I was not and that I had been." In the generated Thackary-30 passage, we see a similar symmetry emerging, but it seems to get "stuck" on the phrase "good deal": "good deal, says Giglio, and that when I was a good deal of me...I shall be a good deal of men of my familiarity and a good deal." The Thackaray-60 passage seems to improve on this, and we have the construction involving the "profession…and the second profession."

Two key, interrelated questions emerge from this comparison. One, why does the RNN seem to capture Dickens better than Thackeray, two authors of the same era (who indeed were friends and oftentimes rivals) writing about similar subjects in styles that, while not identical, share many similarities with one another? Second, why should extending the sequence length (what I have referred to as our agents' attention span) improve the output of the Thackeray RNNs but have little effect on the Dickens RNNs?

Approaching these questions from a literary critical perspective, I propose that much of what makes Thackeray difficult for RNNs from a computational standpoint is the "dry wit" I mentioned at the outset of this subsection. Taking *The Luck of Barry Lyndon* as our working example, much of this wit manifests itself through the first-person narration of Redmond Barry

[9] Unlike the generated Thackeray passages, it made little difference with Dickens whether the sequence length was set to 30 or 60 so I did not include the latter output of "Dickens-60." By and large, I found that increasing the sequence length did not improve the performance of my trained RNN output; Thackeray was an exception. This is my motivation for choosing these two authors here.

himself. Barry is not only a sarcastic narrator, but an unreliable one. In the instances where we sense these narrative characteristics most forcefully—take the opening of the above passage when Redmond states "About this time, it must be premised, the United Kingdom was in a state of great excitement from the threat generally credited of a French invasion"—one can feel the sarcasm dripping of the page.

Dickens, on the other hand, is a *serious* writer. Indeed, in what is known as the "Dignity of Literature" debates, the two authors came to a bitter disagreement over Thackeray's blend of humor and subtly veiled satire versus Dickens' more forthright engagement with societal problems via a style of more concrete characterizations and narrative arcs.[10] Stylistically, this difference manifests itself in Thackeray's tangled first-person narrative locutions, the primary device by which the reader finds both humor and insight into Redmond Barry. The prose is structured by Thackeray, for the benefit of the reader, to stage Barry's convoluted, flawed, or tautological sense of the world. On the other hand, *A Tale of Two Cities* is related in the omniscient third-person voice. Dickens is telling us a story; Thackeray invites us to witness Redmond Barry's telling of his own story.

This difference is important computationally because a primary side-effect of Thackeray's narrative strategy are very long sentences. We saw in our example of the parenthetical clause that an RNN needs relatively short sequences to learn effectively. While it is not at all clear that what we call a "structure" in stylistics is at all equivalent to what a character based RNN considers a "structure," what is clear is that an RNN requires repetition to

[10] See Flynn, Michael J. "*Pendennis*, *Copperfield*, and the Debate on the "Dignity of Literature"." *Dickens Studies Annual:* Essays on Victorian Fiction 41, no. 1 (2010): 151-89.

learn properly, and that this repetition must occur within the bounds of its extended horizon of memory. More subtly, RNNs seem to work best on declarative utterances, which occur much more frequently in Dickens-as-storyteller. Even to a reader unfamiliar with the formal qualities of stylistics, it is quite clear that "It was the best of times, it was the worst of times…" has a cadence to it, a cadence which emerges *because* it is declarative. The cadence of Barry Lyndon is much more diffuse precisely *because* it is attempting to capture an often-disorganized thought process. Because of this diffusion of repeated structures, we see an improvement when we increase the "attention span" of the RNN trained on Thackeray.

In Thackeray, then, we have a case in which an RNN configured to consider longer sequences yields better (but still relatively poor) results. To understand exactly why the algorithm performs slightly better with this configuration, as well as what "attention spans" mean from a technical perspective, we must explore at a lower level how an RNN (and indeed, any ML system) learns. As we saw in the prior section, we need to "train" the RNN over many epochs, and each epoch of training generally reinforces patterns that allow the RNN to refine their predictions and produce more coherent output. Computationally, when an RNN learns it performs a process which is known as "gradient descent."[11] Gradient descent is a sophisticated variant of the process of numerical "root finding," a widely used class of algorithms that are found even in everyday software such as Microsoft Excel. In Excel, the user has access to a

[11] For a technical overview of gradient descent in ML training, see Bottou, Léon. "Large-Scale Machine Learning with Stochastic Gradient Descent." *Statistical Learning and Data Science*, 2011, 17–25. https://doi.org/10.1201/b11429-4.

built-in function called “goalseek,” which employs a simple root-finding algorithm to find the lowest or highest point on a mathematically defined curve.

Root-finding algorithms can be incredibly complex, especially when dealing in spaces higher than two dimensions. To understand why, let us construct another metaphor to illustrate a simple root finder. Suppose you are trapped in a landscape of hills and valleys and you wish to find the highest peak. Because you begin in a valley, you cannot see where that peak might be. A reasonable way to approach this problem is to look directly downward at the ground and take one step in a chosen direction: north, south, east or west. At each step, you consider your four options and always choose to take a step in the direction that has the steepest grade upward.

What could go wrong? If you are always traversing upward, why shouldn’t you always find the highest peak? The answer is that you could reach a peak where, by definition, all your directional choices point down. You conclude you are finished, and you have indeed found a peak, but it is not necessarily the *highest* peak in your landscape of hills and valleys. There could be a very much taller peak half a mile away, but you can’t see it or figure out how to get there, because you are only considering the ground one step in front of you. You have found what is called in mathematical terms a *local maxima*.

When RNNs are taking these same steps in their training epochs, they are very susceptible to being caught in local maxima or minima, which, in the world of ML, are called “exploding” and “vanishing” gradients. When caught in an exploding or vanishing gradient, our RNN algorithms cannot make any progress, because they cannot find a character that will

improve its results. It has reached a peak or trough where no choice of character will improve its accuracy.

We see a manifestation of an exploding/vanishing gradient (or local maxima/minima) in the first sample in Thackeray-30 in which the output continues to repeat the phrase ""good deal": "good deal, says Giglio, and that when I was a good deal of me...I shall be a good deal of men of my familiarity and a good deal." One contributing factor to Thackeray's loquaciousness is his tendency to use relatively complex and variegated phrases as his conjunctive transitions:

> For, if the truth must be told, I had made very deep love to her during my stay under her roof; as is always my way with women, of whatever age or degree of beauty. To a man who has to make his way in the world, these dear girls can always be useful in one fashion or another; never mind, if they repel your passion; at any rate, they are not offended with your declaration of it, and only look upon you with more favourable eyes in consequence of your misfortune.[12]

"If the truth must be told," "never mind," "at any rate," are not strictly necessary to conjoin Redmond Barry's narration; the semicolons would suffice for that purpose. But in this passage, these throw-away phrases serve to convey Barry's braggadocio-*cum*-confession under duress ("if truth must be told") and to stage a crass indifference ("never mind," "at any rate") to the women he makes "deep love" to, a rhetorical strategy that could indicate genuine crassness or conceal a deeper affective connection than he cares to express.

I suggest that the shorter memory of the Thackeray-30 cannot look past these complex conjunctions because it is either trapped in a "valley" (or vanishing gradient) where it cannot

[12] Thackeray, William Makepeace. *The Luck of Barry Lyndon*. Edited by Walter Jerrold, 1844. Project Gutenberg, www.gutenberg.org/files/4558/4558-0.txt.

cognize these as the transitional elements they are, or it is isolated on a "peak" (or exploding gradient) where it cognizes a particular conjunctive phrase (here, a "good deal") as something much more fundamental than it really is, i.e., a formal grammatical structure like a semicolon or an "and." The Thackaray-60 RNN is allowed to metaphorically pick its head up a little further, and as such has a greater chance of recognizing when it is caught in a local maxima or minima.

With this demonstration of the tuning of an RNN's hyperparameter of sequence length, we can begin to formulate at least two distinct modalities of attention. One modality, belonging to the nonconscious cognition within the RNN algorithm is simply a linear attentiveness, an extension of sequential memory that can prevent the RNN from succumbing to exploding or vanishing gradients. But in our capacity as literary critics, we have an additional mode of attentiveness. In Dickens, we identify the "storytelling" mode of the third-person universal point of view, and consciously focus our attention on higher-order structures such as character and plot. In Thackeray, we eventually come to realize that Redmond Barry is an unreliable narrator, and that the story we need to pay attention to is not primarily about *what* he is telling us but *how* he is telling it to us. Ironically, I believe that this conscious realization leads our attention in the opposite direction as the RNN: we broaden our focus in Dickens to comprehend the arc of the plot and the way in which the characters systematically interact with themselves and their social milieu, whereas in Thackeray we narrow our focus to contemplate the how structures and misdirection of narration explicate how an individual relates to the world.

Section 3.3: Technical and Aesthetic Attention

In our examples of RNN tuning, we explored three properties of algorithmic attention: 1) attentiveness to different granularities of information, e.g., character versus word, 2) attention as a learned trait that accretes via exposure to repeated sequences of information and, 3) the effect and importance of varying degrees of attention span. Each of these algorithmic traits has analogues in human cognition: we (consciously or nonconsciously) choose to attend to specific informational flows that are captured by our senses, humans learn that some things are important to attend to in our daily lives, and we, too, possess and allow ourselves varying degrees of attention span.

At some level, then, human attentiveness seems very much like the attention we explored within the RNN. Yet one core difference that immediately presents itself is the human cognitive ability to adjust each of these attentional "knobs," dialing them up or down given ambient, extrinsic feedback. A human observing small patterns of raindrops may shift their attention to the approaching storm on the horizon upon hearing a menacing thunderclap. The human who touches an unexpectedly hot surface will likely infer that this situation merits attention based on a single experiential observation. And, perhaps most problematically from the standpoint of a machine, humans can adjust their attentional span to capture the entirety of what they perceive as a single, self-contained event. This last facet of attention span is problematic because attention is commonly understood as a feedback network: we hear the growl of a potentially threatening animal and shift our attention as a result of this perceived threat. Yet when viewed from the standpoint of an algorithm, we also see that a feedforward component must also be involved: an event can only be cognized *as an event* if it can be

captured by the current duration of attentional span. Modulating attention span to delineate discrete events presents intense difficulties for algorithms but often for humans as well: one might ignore the extended symptoms of a serious health issue while attending immediately to the discomfort of a paper cut.

Attention therefore not only describes a systematic interaction between multiple modes of information processing, but also manifests itself as the organizing temporal structure of informational units themselves. We can effectively unpack this formulation by revisiting the RNN's inability to cognize Thackeray's variegated and verbose clauses—"If the truth must be told," "never mind," "at any rate"—as highly stylized forms of simple conjunctions: "And," "Yet," "But," etc. The RNN could not learn how to position Thackeray's phrases as the conjunctive glue which they provide in the source texts. Yet the RNN performed well in texts from authors such as Dickens and McCarthy, and the transitions between sentences are rendered more effectively. Dickens and McCarthy both tend toward a declarative style: "It was the best of times. It was the worst of times.." in Dickens, "The kid rode…Glanton spat…" in McCarthy. The transitions in Dickens and McCarthy, to the RNN, are simple periods, likely followed by an often repeated, simple subject: "It," "The kid," "Glanton." To cognize a transition between one phrase and another in Dickens or McCarthy, the RNN only needs to pay attention to a local span of tens of characters, and crucially the placement of periods. In the long meandering sentences of Thackeray, transitional states between thoughts and sentences not only occur over a longer attention span, they crucially *require the same attention span* as the normal flow of text within non-transitional formal structures.

Now I maintain that human interpreters come to recognize Thackeray's discursive style, and while this recognition does not preclude readers appreciating this style as part of an aesthetic whole, it nonetheless allows them to cognize these lengthy transitions with *less* attention than other portions of the text. In other words, while readers may initially attend to these transitions as signifying some deeper structure about Redmond Barry's intrinsic evasiveness, they eventually learn that they form a quasi-pattern, a repetitive structure that, once cognized as such, demand a relatively brief attention span. The key difference is that attention span in RNNs is always a function of the length of a sequence of characters, whereas humans truncate or elongate attention spans in reaction to an evolving expectation of what is required to cognize a unit of information. Human readers encounter "at any rate" so often in Thackery as to recognize it as a simple transition, and dwell on it no more so than a normal conjunction. Yet if the same phrase were to appear in McCarthy, readers would likely pause and increase their span of attention until they could cognize what such an atypical phrase might mean within the suddenly increased awareness of the surrounding text.

This adjustment of attentional span must be carried out prior to the other components of attention, as it is the key to determining that a given span of information constitutes an event. Once determined to be an event, a unit of information—whether that information manifests itself as a phrase in a text, the roar of a threating animal, or the embodied experience of a paper cut—can be subject to higher (or lower) order cognitive processes that determine whether and to what extent that event merits attention.

Section 3.4: Conclusion

In this chapter, we constructed several experiments to explore the inner workings of RNNs. These experiments demonstrated various low-level technical facets of the algorithms and parameterizations that would be familiar to a technical audience. By reading the textual output of the RNN as literary critics, we were able to correlate the degrees of success or failure of the algorithm to recapitulate different authors to humanistic assessments of meaningful formal or aesthetic textual qualities. In this cross-disciplinary interaction, certain modes of discourse or authorial idiosyncrasies familiar to literary scholars can be bought into contact with the instantiated code of a functioning RNN. The training and resultant behavior of RNNs represents an emergent technical process that, in its complexity, evades comprehension as a deterministic algorithmic system. Yet by combining a humanistic understanding of the textual output with examples that demonstrated some facets of the low-level implementation of the algorithm, we found a way to peek into the black box, gaining insight as to how these tools work and why they sometimes fail.

This insight allowed us to explore one of the most complex and topical areas of ML research, the concept of attention. In our exploration of RNNs, we have glimpsed their complexity as well as their "unreasonable effectiveness." The questions of *why* an RNN works and why they fail can be attributed, at least in part, to the system's inability to differentiate between different modes of attention. In the RNNs we have explored, attention is given only to single characters, and the only form of attention that can arise is a form of probabilistic expectation given the limited, fixed-length context contained within the algorithm's memory. But in interpretive reading, attention is not a fixed quantity, it expands and contracts in

accordance with our notion of what is important within a text. And furthermore, this human ability to expand and contract spans of attention yields an organizing structure, allowing us to cognize events or textual features as singular phenomena, discrete things that emerge as discrete because they occur within the duration of our attentional focus.

Once again, formulating a key technical concept in humanistic terms yields an understanding of the differences in algorithmic versus human cognition. And in this chapter's greater focus on the technical and experimental, we have glimpsed the complexities of creating and parameterizing code to model something as complex as attention as it occurs in interpretive activity. RNNs, are, indeed, unreasonably effective. To understand an unreasonably effective algorithm, we must take in earnest that their computational processes are, in fact, unreasonable. And when reason fails us, we must look to those human capacities that lay beyond pure reason to come to terms with both the effectiveness and limitations of the tools we have built.

Coda

Reading Algorithms as a project and ideal emerged from my early employment in the University of Chicago's Artificial Intelligence lab. As an undergraduate earning my degrees in Computer Science and Math, I found little space to fit English courses into my schedule—an unfortunate situation for a youth who found much enjoyment and solace in literature of all stripes. When I discovered that the University of Chicago allowed employees to attend graduate courses, I jumped at the opportunity to finally discuss books with fellow bibliophiles in a classroom setting. As one can expect, fifteen minutes into the first discussion I found myself entirely out of my depth, stunned at my classmates' erudition and doubly confounded by the mysterious apparatus of critical theory.

It says much about the quality of my instructors and patience of my peers that I was eventually able to settle into the discussions and begin to develop an understanding of the theoretical constructs. Most compelling to me was learning about the entanglement of linguistic structures in cognition and the (un)conscious. At my day job in the AI lab, I was writing code to help search engines parse text questions as well-formed, grammatical queries. In today's world, I believe that almost everyone who uses Google knows that the box you type in doesn't *understand* what you are asking; the algorithm understands keywords and cleverly transforms these into high-quality results. It is telling that we now enter keywords into that box instead of questions; we modify the way we enter questions because we have some notion that this format is what the algorithm "wants."

Yet pre-Google, it was not clear that keyword-based searches could provide enough context to yield good results, and as such I was assigned the task of programming a tool to "understand" a question phrased in everyday language. As a junior researcher with a fresh degree in Computer Science, I thought this eminently possible. It took those English courses to realize the Sisyphean nature of this task. I had never considered language as a *tool* to understand the world, or interpretation as anything but a conjunction of grammar and syntax that inexorably led to meaning. On the one hand, I despaired that my current AI project could succeed, but on the other, I became fascinated with these deeper facets of language and interpretation.

I bring up this personal anecdote to conclude this project on a mildly polemical note. While modern researchers in AI and ML are nowhere remotely naive as I was as to the complexities inherent in natural language processing, I believe there is a sort of confidence that has emerged as a result of the recent, meteoric rise of "deep" ML algorithms in practical (and ubiquitous) applications such as Siri. This confidence is justified, in part, simply because these algorithms do work, and they are indeed improving. Yet I entertain a nagging suspicion that some measure of this confidence draws from a more occult reservoir. In most cases, these specialists do not know *why* they work. But the algorithms keep improving with various tweaks to the code, and a species of *faith* emerges that there is no limit to this cognitive improvement—nothing, in other words, to prevent the machine from advancing inexorably toward human thought. Now to be sure, an ML specialist knows how the algorithm is coded and how information flows through the system. Yet as we saw in the unreasonably effective RNNs—and what I often encounter in the technical literature—is a deep puzzlement over how

these algorithms collate these massive calculations into results. There is a form of magic here, and I believe that the incomprehensibility of these algorithms, coupled with the fact that they are improving, invites researchers, as well as the general populace, to have faith in that magic.

But higher-order human cognitive facilities contain a similar magic, in the sense that we do not fully understand how they function, yet we continue to be amazed at our capacity to learn and comprehend. What I have hoped to have shown in Reading Algorithms is that humanists possess unique insights into these cognitive facilities. When we can demonstrate qualitatively different modes of cognitive engagement in human activities such as literary interpretation, we can frame these exuberant or (in the worst case) apocalyptic visions of ML in a productive way. This is not to say we should act as continual pessimists, continually pointing out "yes, but can your algorithm do this…" in response to every advancement in ML technology. Rather it is to suggest that if humanists treat these algorithms as serious objects of study, we can explore the limitations of this technology productively, suggesting, as an example, why an application such as Siri might have difficulty with certain modes of linguistic interaction. And along the way we can perhaps restore some degree of appreciation for the depth and complexity of human cognitive facilities.

Alongside simply providing a more realistic framing of ML technology, I have a far more pressing concern. Recall my original AI failure to create a "natural language" search engine and the rise of Google-based keyword searches. Most of us are now accustomed to forming our questions as keywords because we know that this is how Google cognizes our queries. As many philosophers of technology have observed, our tools condition (and perhaps create) certain behavioral and cognitive patterns. ML algorithms represent some of the most complex tools in

history, and if we conflate their cognitive (in)capacities with our ability and desire to make ourselves cognizable to them, we are very much headed toward a merger of human and machine in the worst possible sense, a flattening of cognition that will always tend toward the lowest common denominator. A future, in my estimation, far bleaker than one dominated by Skynet.

I do not want to end on this apocalyptic note, and, indeed believe that many problems can be solved by opening collaborate spaces between humanists and technologists. This is an entirely practical and desirable goal, taken from vantages both inside and outside the academy. From the research and academic side, I can easily imagine a manifestation of DH that would not shy away from opening their algorithmic toolkits for humanistic analysis. Not necessarily as code, but as systems of processing information that can be analogized and conceptually inhabited by readers of texts. This would allow humanists to understand how an algorithm cognizes information, allowing them to mobilize their own domain knowledge to help technologists debug and inhabit their own code. From the side of the commercial, this collaboration can be justified with the simplest of capitalistic arguments: it can make money. Imagine an application like Grammarly that was aware of the discursive context in which the user is writing. Legal discourse contains innumerable formal features that distinguishes it from a back-and-forth volley of emails between middle managers, and an ML system that could tune its suggestions accordingly would be far more effective than a naive one-size-fits-all approach. Similarly, one could imagine a less irritating form of Grammarly that would learn your style and allow you to write with some measure of individuality, yet still catch obvious mistakes or universally grating locutions. Both tasks would require substantial collaboration between

humanists and technologists, and those who would excel in this professional milieu would be those who could most effectively navigate these boundaries.

As a junior programmer in the AI lab, I was blissfully ignorant of these boundaries, only vaguely aware of the academic divide between the humanities and the sciences. And my fortunate first steps into those English classrooms heavily influenced my career as a technologist. Likewise, my interest in interpretive theory was sparked and continues to be influenced by a technologist's mechanistic view of text as pure data. I do not feel as if I am at all unique in this regard. I believe that many technologists—especially those dedicated to linguistic-based ML—would be as delighted as I was when I first grasped the depth and complexity of humanistic interpretive processes.

But to gain pedagogical traction, the idea of collaboration is necessary. DH has been useful to introduce some notion of how computational methods can be integrated into literary study, but it is a one-way street. A certain type of literary critic might be interested in a statistical analysis of genre theory, but I would think that problem would have limited appeal to a dedicated ML researcher. What would interest an ML researcher is to start with difficult, real-world problems that *have not* been solved, such as the examples I have given around sequence, context, and attention. These problems will not be solved in the near future, and the output of the collaboration will not be a neat set of statistically significant results and appealing charts. What will result is a mutual understanding not only of the difficulty of the problem, but *why* it is difficult. As a young technologist in those first English courses, this understanding was exhilarating. It is my hope that "Reading Algorithms" demonstrates how this same enthusiasm can be extended to others hoping to bridge technology and humanism.