



BIROn - Birkbeck Institutional Research Online

Eve, Martin Paul (2020) Distance and Depth. Stanford University Press Blog
,

Downloaded from: <http://eprints.bbk.ac.uk/30723/>

Usage Guidelines:

Please refer to usage guidelines at <http://eprints.bbk.ac.uk/policies.html> or alternatively
contact lib-eprints@bbk.ac.uk.

[SUP Home](#)

[Blog Archives](#)

[About the Blog](#)

Distance and Depth

Understanding the intersect of literary text and computers.

MARTIN PAUL EVE



Free for commercial use. Via [Pixabay](#).

One of the ways in which literary studies recurrently defends itself is through the idea of *critical attention*. English literature, as a field, is sold as allowing students to move from eddies of formalist close textual attention to the tidal fluxes of large-scale literary history, all while developing their thinking within politico-theoretical streams that extend the possibility of progressive real-world change. These oscillations between theory, archive, and formalism sometimes form the bases for scholarly contention: ‘what is the *proper* object of literary studies?’ For students of the discipline, though, such variety often coheres into a dynamic and stimulating field that does, indeed, teach critical thinking skills and can foster a life-long love of reading.

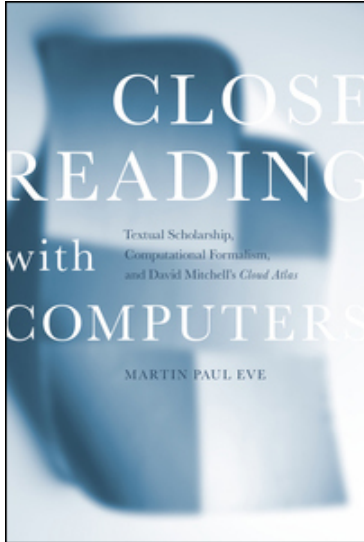
‘To err is human’, the [oft-misattributed quotation runs](#), ‘but to really foul things up you need a computer’. And ‘computational literary studies’ or ‘digital literary studies’ has [attracted some flack](#) for fouling things up, to put it mildly. Conventionally, approaches in this space towards literary texts have been critiqued as ‘bad’ [stylistics/formalism](#); as [a-theoretical/anti-progressive](#); or as [historical mis-readings](#). The apparently scientific incursion of computational approaches evokes a fear for the historical, textual, and political sensitivities on which literary studies has staked its claims of critical thought. The fact that digital humanities seems, to many, to attract funding only serves to confirm its misplacement in literary studies departments.

Many arguments for digital approaches to the study of literature, though, rest on the problems of canon and selection (which, of course, are overtly political). More works of fiction are published every year than it is possible to read in a lifetime. Our judgements on the worth of fiction, both new and old, are conditioned by a set of market practices that limit our understanding of [true mutations in writing](#) over time. In this light, recent books by [Ted Underwood](#), [Andrew Piper](#), and others have built on the traditions of so-called [distant reading](#) but also predictive modelling in order better to understand the larger scale sweeps of literary history, without resorting to periodizing description. Charting continuous change rather than distinct epistemic and aesthetic breaks, such methods examine computationally the effect that linguistic mutations have upon our understandings of literary history.

Using software as a tool to explore texts can lead us to new literary insight, if we retain a vibrant critical, theoretical, historical, and interpretative imagination.

There is a second strand of digital work in literary studies, though, that underpins such an approach to ‘distance’: text analysis. Often operating at smaller scales, for distance should not be opposed depth, [such work](#) seeks to [extract linguistic features](#) from individual texts or authors and then to visualize or otherwise present the results. This can be of use as the repetitious nature of examining, say, foregroundedness (in which a textual feature stands out with prominence) as opposed to its actual quantitative presence (how often it really occurs) is beyond the patience of most readers. Yet distant approaches at scale rely on this microscopic approach working, the idea being that if we can accurately measure and computationally discern features at the atomic level we can then scale this up to work on unknown texts and corpora.

I have been interested, then, in the ways in which computational methods might help us to understand the formal strategies that authors use to create specific effects at the level of a single novel. For example, if an author wishes to create the impression that a text was written in the nineteenth century, from within the twenty-first, what does s/he do? In this example, looking at David Mitchell’s 2004 genre-bending novel, *Cloud Atlas*, I first wrote a computer program that could identify anachronism. Revealingly, this did *not* sit at the root of Mitchell’s stylistic achievement in that novel, despite the fact that Mitchell’s use of language is almost entirely accurate; the only three terms that I can say with any certainty that sit out of their time were ‘spillage’, ‘Latino’, and ‘lazy-eye’.



[Close Reading with Computers](#)» is the first monograph to bring a range of computational methods to bear on a single novel in a sustained fashion, it focuses on the award-winning and genre-bending *Cloud Atlas* (2004).

set of methods for exploring aesthetic artforms.

[Start reading Close Reading with Computers »](#)

Martin Paul Eve is Professor of Literature, Technology and Publishing at Birkbeck, University of London. He holds a Ph.D. from the University of Sussex and is the author of [Close Reading with Computers](#) and four other books. In 2019 Martin was awarded the Philip Leverhulme Prize for his literary studies work.

Posted on January 29, 2020 in [Literary Theory](#) | [Permalink](#)

Tags: American Literary Studies, British Literary Studies, Eve, General Criticism and Theory, Literary Studies

Tweet | Like 15

Comments

Verify your Comment

Previewing your Comment

Posted by: |

This is only a preview. Your comment has not yet been posted.



Your comment could not be posted. Error type:

Your comment has been posted. [Post another comment](#)

The letters and numbers you entered did not match the image. Please try again.

However, the respective racial and pejorative nature of two of these terms sparked, for me, another line of inquiry: was it possible that at least part of Mitchell’s stylistic imaginary works by exploiting our positivist conceptions that racial descriptors and disability slang would saturate nineteenth-century prose and that such terms *would* be available to an author of that period? Compared to a contemporary corpus of writing I found that, yes, Mitchell does use colonial terms of racial abuse, for instance, with a much greater prevalence than a broader contemporary corpus.

Yet, noting such aesthetic features only tells us so much, regardless of how interesting it may be in its own right. Without the multi-directional pulls of theoretical understanding and literary history, such feature spotting fails to show us *why* textual effects translate into meaning. This is why my work is titled [Close Reading with Computers](#)», for despite the above fears, it is not through computation and digital methods that we find meaning, but in our human engagement with words and things, with language and themes, within our own contexts of reading. Using software as a tool to explore texts can lead us to new literary insight, if we retain a vibrant critical, theoretical, historical, and interpretative imagination.

Under such conditions, it then becomes clear that repetitive counting and computational methods can tell us more than mere frequency. It is not true, as [Timothy Brennan tried to argue](#), that counting the word ‘whale’ in *Moby-Dick* can tell us no more than the number of times that the word ‘whale’ occurs in *Moby-Dick*. Just as traditional approaches to close reading since the Modernist era of their birth work backwards from textual evidence to hermeneutic stance, quantitative data, garnered with the use of computation, are another source that can be so used (with a pedigree stretching back at least to Vernon Lee, it might be added).

This does not mean that all literary scholars should ‘go digital’ or even quantitative. It just means that, once more, there is a diversity of practice in our disciplinary community, alongside a new