The Beginnings of Content Analysis:

From the General Inquirer to Sally Sedelow

Geoffrey Rockwell and Stéfan Sinclair

In a 1968 “review symposium” three reviewers savaged the content analysis tool *General Inquirer* (GI) and associated book *The General Inquirer: A Computer Approach to Content Analysis* (by Stone et al. 1966). As one of the reviewers, James D. Merriman, put it,

> The dangers of inserting our hypotheses into the data and then pulling them back out like rabbits from a hat are really at issue. (Kadushin et al. 1968, 194))

Despite this triple review of a tool, the *General Inquirer* had supporters and continued to mentioned in *Computers in the Humanities* over 40 times right up to an appearance in a bibliography of an article in 2004. The first sustained defense of the GI came less than a year later from George Psathas (1969) who wrote a positive review in response titled, "The General Inquirer: Useful or Not?" Needless to say, he felt it was useful if used properly.

From the safe distance of another millennium we can ask why was this tool and its associated method of content analysis so controversial? The answer lies partly in differences between how the humanities and social sciences approached the computer processing of text and that is what this paper will be about, the difference between a diagnostic use and hermeneutical use of computers for analyzing texts. In the paper we will:

- Start by discussing content analysis and how it was implemented in the *General Inquirer*. We will introduce a replication of the GI in Spiral, a new literate programming environment we have developed.
- Then we discuss the *Via* project by Sally Sedelow that implemented an alternative and more hermeneutical version of content analysis that she sometimes called stylistic analysis.
- We will conclude by

Content Analysis and the *General Inquirer*

The *General Inquirer* (GI) was a foundational attempt to use computers to get at the *content* of a text methodically rather than simply automate concording or use the computer to characterize the style of a text. The *General Inquirer* was initially developed by Stone and colleagues in 1961 for the IBM 7090 at Harvard and later adapted for smaller mainframes.

Content analysis was and still is, one of the most important uses of computers in the social sciences (Neuendorf 2017) and one of the most problematic applications of text analysis for humanists used to the subtleties of interpretations drawn from the text rather than from scientific methods. The GI was the first of a long line of tools that automated content analysis

as a method through the use dictionaries of categories of words leading up to current tools like LIWC2015 (Linguistic Inquiry and Word Count) which bills itself on its web site as "the gold standard in computerized text analysis." (liwc.wpengine.com) Such tools use dictionaries for everything from diagnosis to sentiment analysis.

How does Content Analysis work?
In its implementation as documented in the 1968 *User's Manual for the General Inquirer*, the system does the following.

- It operates on a set of texts in batch mode.
- TAGGING: It deals with each text sentence by sentence tagging the sentence by examining each word.
    - The words can be stemmed by removing common endings
    - The words are checked against a category dictionary of word stems
        - Common words are then disambiguated following rules from the dictionary
        - If the word stem (or word sense) is in a category then the sentence is tagged as belonging to that category
    - Optional "Sentence-Summary" tests can be applied after the words of the sentence are processed. These tests look for patterns of tags and/or words and then add or remove tags. For example, a new tag can be added if a particular combination of category tag and word are found in the sentence.
- The results can be output in different forms for different types of post-processing like a:
    - Listing of sentences and tags
    - A "Leftover List" of words that didn't match a category
    - A "Tag-Tally" of the percentages of category tags per document
    - A graph of categories
    - A KWIC Index

I've left some details out, especially around the markup of the entry text and structure of the dictionaries. Current tools like LIWC2015 are simpler and just give you tallies of the categories. Here you can see the output of LIWC2015.

<Show GI notebook>
Here is Spiral notebook that replicates the simple tag tallying process of GI. We use the dictionaries of categories that can be downloaded from the General Inquirer site.[1] The Basic Spreadsheet is what was used by GI and combines the words in the Harvard ("H4") and Lasswell ("Lvd") or both ("H4Lvd") dictionaries. It has a total of 182 categories many of which have overlap. For example, there is a Negative category, an earlier Ngtv category and a subset of Hostile.[2]

[1] See <http://www.wjh.harvard.edu/~inquirer/spreadsheet_guide.htm>, Accessed May 21, 2017.
[2] For explanations of the categories see, <http://www.wjh.harvard.edu/~inquirer/homecat.htm>, Accessed May 21, 2017.

Where our notebook varies is that we have better lemmatization algorithms, but we have also left out some of the complexity of what the GI did.

What are the interpretative assumptions of Content Analysis?

> The program is an "objective" description of the content analysis process, it must operate on "manifest" features of the text, and it can be designed to yield carefully counted "quantitative" results. (Berelson 1952, 241)

Content Analysis has its roots in the analysis of communications in the social and behavioural sciences going back to pioneering work done manually during WW II on propaganda. Stone drew on Bernard Berelson who used the phrase for the method in his influential 1952 book, *Content Analysis in Communications Research*. Berenson summarizes some of the features of CA thus:

- It is about the "manifest" content of communications, not the motives, the (poetic) effects, or the implied content.
- It is an "objective" method in the sense that the processes and categories are defined so that results are reproducible.
- It is quantitative in that "numerical frequencies" are "assigned to occurrence of the analytic categories." (p. 17)

We should add that one of the strengths of CA is that it can be performed on any type of communication from works of art to TV news programs. The computer may not be able to automatically tag such texts, but people can, if they are given a sufficiently robust rubric.

Also interesting are the assumptions underlying CA that Berelson identifies; interesting because they are true of interpretation in general, whether computer-assisted or not.

- That knowledge of the content can support inferences about intentions and effects. (p. 18)
- That CA "assumes the study of the manifest content is meaningful." (p. 19)
- And finally, CA assumes that quantification is meaningful even if by quantification we talk about "more" or "less" of some theme.

Content Analysis and the Humanities

Given how popular CA is in the social sciences, why is it not used then in the humanities? The main reason is because CA is not hermeneutical in the sense of drawing a hermeneutic from the subject of study. CA starts with a theory about content in a phenomenon and uses that to classify the instances of the phenomenon, the texts. CA starts with either a dictionary of categories or, if you do it manually, a rubric and instructions on how to read the works being classified. The theory is then validated by the tagging rather than developed in the reading. It uses the theory to diagnose the works instead of letting the work guide its interpretation. This

relationship to theory is what makes it scientific in the sense that it should be objective and reproducible. Humanist interpretation is ideographic – we treat each human work as unique and let it guide us in our interpretation. Scientific analysis is nomothetic; it tries to develop general laws that can be applied to each work. This plays out in the choice of texts and how we work with them. The humanist works with exemplary works (of art) worthy of individual interpretation; the social scientist works with documentation of a phenomenon like interviews and treats them as a means to an end.

> "During these 25 years, the social sciences have witnessed a shift away from grand theories to more practical, grounded research – a shift that is somewhat reflected in the choice of thematic text analysis procedures." p. 47

In a late essay from a book published in 1997, Stone recognized the difference and imagined tools that could suit the different approaches. We will now turn to Sally Sedelow's work, as she was one of the first to imagine more grounded text analysis.

Sally Sedelow and *Via*

Less well known than the *General Inquirer*, but also influential in other ways was Sally Sedelow's VIA tool that took a different approach to content analysis using a thesaurus rather than a dictionary of theoretically developed categories. Sedelow, who has been virtually forgotten in the digital humanities, despite her importance as a developer and teacher, had an English PhD. from Bryn Mawr and had done research for the Systems Development Corporation, a spin-off of RAND founded in 1955 and at the time the largest software company in the world (Campbell-Kelly 2003). With support from the Office of Naval Research she developed the *VIA* system which took a humanistic and interpretative approach to content analysis. VIA was to become the basis for her student John B. Smith's first toolset called RATS (Smith 1972) which in turn evolved in ARRAS, the first interactive and interpretative text analysis environment, and a model for tools from TACT to Voyant. That Sedelow and *VIA* have been forgotten says something about the lack of attention given to the scholarship that takes place in tool development, but that is another story.

What distinguished *Via* was its use of a thesaurus in text analysis to develop a custom dictionary grounded in the text. Simply compared, while the GI used a prepared dictionary of categories to *diagnose* concepts in a text, VIA used a thesaurus to *propose* candidate themes by uncovering clusters of synonyms from the text. The themes could be edited and then used to *interpret* a text through its own categories (Sedelow 1964).

<Show Spiral notebook>
Here you can see a Spiral notebook that implements our interpretation of what Sedelow was experimenting with in the 1960s and 70s.

- The tool typically operates on a single text.
- The text is tokenized on words.
- The words are lightly stemmed.

- They are then looked up in the thesaurus and the frequency of different thesaurus categories is counted
- The use can then get a report that shows the high-frequency categories and the words in the categories that appear in the text.
- Sedelow was also experimenting with multilevel thesaurae where you can see the network of categories appearing in the text.
- This would allow the user to edit a custom dictionary or thesaurus with which to interpret the text closely or with which to interpret a related text.

How do these approaches compare? Both approaches have limitations that developers are still struggling with, including the problem of disambiguating homographs, problems of meaning in context, and problems with predetermined categories whether theoretically developed or determined by a thesaurus.

> Looking back to when text analysis was carried out on mainframe computers, it may well be that part of the appeal of general theories was that they provided a relatively broadside mapping of the text being studies. Inasmuch as researchers had infrequent access to the mainframe, they wanted to take away as much information as possible. Indeed, huge, unnecessarily large printouts characterized computer use at that time. By comparison, desktop computing encourages a more grounded approach to research much like detective work, zeroing in on key evidence rather than making broadside passes over data. p. 52-3

We believe, as Sedelow did, that the dictionaries/thesaurae should be thought of as interpretations themselves, but interpretations that can be replicated (Sedelow 1965). We can say that they aim to be ontologically objective while being epistemically subjective, to use Searle's distinction.

We also agree with where Stone was headed late in life. As he pointed out, the chronotope (access and pace of use) of the mainframe encouraged the development of tools that could be run once generating long printout reports for careful reading and interpretation without the computer. This in turn encouraged a batch theory approach where you start with a programmable theory and apply it to a batch of documents. Sedelow saw the potential of a more interactive use of the computer where you can iteratively intervene in the process customizing the theory during interpretation. In short, the type of computing influenced the application of theory instantiated in the tool which then influenced the interpretation itself.

Spiraling into Conclusion

If the form of the computing influences the interpretation then we need to spiral back and reflect on our tools of interpretation by reflection. We will conclude with a brief discussion of what Spiral is and how we are thinking-through it.

Spiral is a literate programming environment (Knuth1984) inspired by notebook programming environments like Mathematica and Jupyter (Python.) In fact, we started the larger project of replicating historically important methods in Mathematica and Jupyter in order to prototype what we wanted for Spiral.

Spiral, which is still in Beta, is built on Voyant so you can combine blocks of text, blocks of code in Javascript, and interactive Voyant panels all laid out in a notebook. It is this integration of interactive Voyant panels that distinguishes Spiral from other literate programming environments. Spiral can be thought of as Voyant notebooks, giving users a way to extend Voyant with functionality programmed in JavaScript. But as this we hope this paper has shown, the notebook model encourages another way of doing text analysis where you document your analysis as you go in a notebook, something you can't do with Voyant easily, given how interactive it is. The notebook forces one to narrate what you are doing, if only by dropping a sequence of panels with different sorts of results. And, as we learned from Sedelow, it also gives you the opportunity to intervene in the process, editing intermediate results to better ground the analysis. To conclude:

- The notebook programming model encourages a certain form of openness where reflection and code are open to view, editing and reuse along with results. The code is no longer hidden away, but becomes part of the interface. This can, of course, be distracting. No one wants to see the code of a word processor being executed while processing words, but where the code instantiates an analytical method, it exposes the instantiation.
- This takes a step further the humanistic insight of Sedelow's *Via,* that the dictionaries of categories are part of the interpretation and should therefore be editable, by recognizing that the code is also part of the interpretation and should therefore also be open, editable, and interpretable.
- In so far as the code and dictionaries instantiate a theory of interpretation, Spiral, like other notebook environments, opens the theory of code to interpretative manipulation. With notebooks you can experiment with our replications – trying them over your texts. In this trying we believe there is a uniquely humanist form of re-plying that keeps open through playful iteration the insights of others where other ways of doing the digital humanities have forgotten Stone and Sedelow's contributions.

Afterword

Finally, and as an afterword, this way of exposing code, has pedagogical applications. To expose the thinking through code is one way of teaching it. For this reason we have followed Sedelow's footsteps from development to instruction and are developing an Art of Literary Text Analysis site on Github in Jupyter (Python) now Spiral.

That said, we need to recognize how replication can also hide through omission. Using a different programming language, different libraries, and having different aims means that replication, as we have practiced it, is not the same as faithful recapitulation. The openness of a

notebook invites exploration through replication, but, like any translation, it can misinterpret by, among other things, being open for examination in a way the original may not have been.

References

Tools and Links

Spiral Notebook implementation of
https://voyant-tools.org/spiral/?input=https://raw.githubusercontent.com/sgsinclair/epistemologica/master/spiral/GeneralInquirer.json

https://voyant-tools.org/spiral/c4c5c03007034c43a8829c7c38a19084? - Geoffrey's Version

Spiral Notebook implementation of Sedelow's *Via*:
http://beta.voyant-tools.org/spiral/experiments/Via

http://beta.voyant-tools.org/spiral/0a37e3623660c8c7c9d7392e18fc4adc

Voyant-Tools implementation of *Via:*
http://beta.voyant-tools.org/docs/#!/guide/via

https://voyant-tools.org/spiral/alta

Literature

Berelson, B. (1952). *Content Analysis in Communication Research*. Glencoe, Illinois, The Free Press.

Campbell-Kelly, M. (2003). *From Airline Reservations to Sonic the Hedgehog: a History of the Software Industry*. Cambridge, MA, MIT Press.

Kadushin, C., et al. (1968). "Literary Analysis with the Aid of the Computer: Review Symposium." *Computers and the Humanities.* 2:4. 177-202.

Knuth, D. E. (1984). "Literate Programming." *The Computer Journal.* 27:2: 97-111.

Neuendorf, K. A. (2017). *The Content Analysis Guidebook*. Kindle Edition. Los Angeles, Sage.

Psathas, G. (1969). "The General Inquirer: Useful or Not?" *Computers and the Humanities.* 3:3. 163-174.

Sedelow, S. Y., et al. (1964). "Some Parameters of Computational Stylistics: Computer Aids to the Use of Traditional Categories in Stylistic Analysis." Proceedings of the IBM Literary Data Processing Conference. 211-229.

Sedelow, S. Y. (1965, February). "Learning the New Methodologies." Paper presented at the *AREA Conference*. 8 pages.

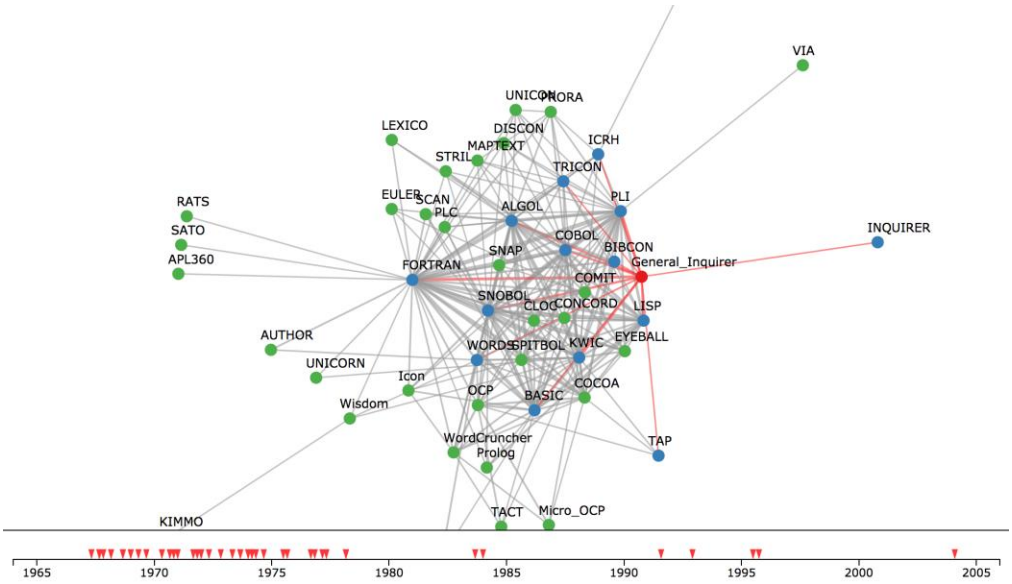
Smith, J. B. (1972). "RATS: A Middle-Level Text Utility System." *Computers and the Humanities.* 6:8. 277-283.

Stone, P. J. and E. B. Hunt (1963). “A Computer Approach to Content Analysis: Studies Using the General Inquirer.” AFIPS '63 (Spring) Proceedings of the May 21-23, 1963, spring joint computer conference, Detroit, Michigan, ACM.

Stone, P. J., et al. (1966). *The General Inquirer: A Computer Approach to Content Analysis*. Cambridge, Massachusetts, MIT Press.

Stone, P. J. and Cambridge Computer Associates, Inc. (1968). *User's Manual for the General Inquirer*. Cambridge, Massachusetts, MIT Press.

Stone, P. J. (1997). “Thematic Text Analysis: New Agendas for Analyzing Text Content.” *Text Analysis for the Social Sciences: Methods for Drawing Statistical Inferences from Texts and Transcripts*. Ed. C. W. Roberts. Mahwah, New Jersey, Lawrence Erlbaum.



Social Network of Tools in Chum (http://cloud.tapor.ca/viz/network/)