Theoretical Grounding for Computer Assisted Scholarly Text Reading (CASTR)

Jean-Guy Meunier, Pierre Poirier, Jean Danis, & Nicolas Payette
Université du Québec à Montréal

Scholarly and Research Communication
VOLUME 3 / ISSUE 2 / 2012

Abstract

Digital humanities technology has mainly focused its development on scholarly text digitalization and text analysis. It is only recently that attention has been paid to the activity of reading in a computerized environment. Some main causes of this have been the advent of the e-book but more importantly the massive enterprise of text digitalization (such as Gallica, Google Books, World Wide Library, and others).

In this article, we analyze, in a very exploratory manner, three main dimensions of computer assisted scholarly reading of text: the cognitive, the computational, and the software dimension. The cognitive dimension of scholarly reading pertains not to the nature of reading as a psychological activity but to the complex interpretative act of going through argumentations, narrations, descriptions, demonstrations, dialogues, themes, etc. that are contained in a text.

Keywords

Reading; Textual analysis; Professional reading environments

Jean-Guy Meunier is a Professor in the Department of Philosophy at Université du Québec à Montréal, 320 Rue Sainte-Catherine Est, Montréal, PQ, Canada H2X 1L7. Email: meunier.jean-guy@uqam.ca .

Pierre Poirier is a Professor in the Department of Philosophy at Université du Québec à Montréal, 320 Rue Sainte-Catherine Est, Montréal, QC, Canada H2X 1L7. He is also Director at Institut des sciences cognitives (ISC). Email: poirier.pierre@uqam.ca .

Jean Danis is a Researcher, LANCI, Université du Québec à Montréal, 320 Rue Sainte-Catherine Est, Montréal, PQ, Canada H2X 1L7. Contact: http://www.lanci.uqam.ca .

Nicolas Payette is a Lecturer in the Department of Philosophy at Université du Québec à Montréal, 320 Rue Sainte-Catherine Est, Montréal, PQ, Canada H2X 1L7. Email: payette.nicolas@uqam.ca .

CCSP Press
Scholarly and Research Communication
Volume 3, Issue 2, Article ID 020116, 20 pages
Journal URL: www.src-online.ca
Received August 17, 2011, Accepted November 15, 2011, Published August 15, 2012

Meunier, Jean-Guy, Poirier, Pierre, Danis, Jean, & Payette, Nicolas. (2012). Theoretical Grounding for Computer Assisted Scholarly Text Reading (CASTR). *Scholarly and Research Communication, 3*(2): 020116, 20 pp.

The INKE Research Group comprises over 35 researchers (and their research assistants and postdoctoral fellows) at more than 20 universities in Canada, England, the United States, and Ireland, and across 20 partners in the public and private sectors. INKE is a large-scale, long-term, interdisciplinary project to study the future of books and reading, supported by the Social Sciences and Humanities Research Council of Canada as well as contributions from participating universities and partners, and bringing together activities associated with book history and textual scholarship; user experience studies; interface design; and prototyping of digital reading environments.

Introduction

In its development, digital humanitie technology has mainly focused on scholarly text editions, dissemination, and analysis. Under the advent of the e-Book and the pressure of massive text digitalization initiatives (such as Gallica, Google Books, JSTOR, Internet Archive, and others) these technologies have started to focus on one of the obvious, but highly modified, activities that it imposes upon experts in the domain: reading texts. An important question has risen: in this new computer environment, what happens to the classical reading practices that scholars have developed, refined, and consolidated within their disciplinary environment?

It is becoming evident that reading practices in both the humanities and the scientific community are suddenly confronting this new technology's impacts. New technology radically changes the access both to the physical carriers of texts and their content. It offers news forms of editions (publishing), dissemination (distributing), new types of manipulations such as visualizing, scrolling, navigating, and new modes of parsing, commenting, annotating, synthesizing, and analyzing.

The traditional meditative attitude for reading a particular text is now washed over by an ocean of other texts that are immediately accessible, and perhaps unconsciously intervene in an expert reading, such as peripheral literature, reviews, commentaries, scholarly interventions, worldwide libraries, and so on. In other words, the classical, easygoing, entrusted text manipulation that scholars have mastered so thoroughly until now is being deeply modified. Expert reading is now different. Technology has appeared to assist in this new type of text reading within a digital ocean. Here we intend to explore some of the theoretical groundings of this specific technology that we call *Computer Assisted Scholarly Text Reading*, or CASTR.

Methodology

A preliminary question

A preliminary question is often raised in the exploration of CASTR technology: "Can a computer be of any help in reading a text?" This is an important question for many expert readers in the humanities and social sciences. Educated in the hermeneutic tradition, and often in reaction to a positivist attitude, these readers offer heavy resistance to the use of the computer as a tool for expert reading of texts.

For these critics, reading is essentially a human, if not a personal interpretative activity, applied to a special type of semiotic system: a text. The computational paradigm is now here on the horizon. Therefore, computers will never be able to read texts or produce

Meunier, Jean-Guy, Poirier, Pierre, Danis, Jean, & Payette, Nicolas. (2012). Theoretical Grounding for Computer Assisted Scholarly Text Reading (CASTR). *Scholarly and Research Communication, 3*(2): 020116, 20 pp.

an expert reading. If they can assist a reader, it will only be with menial tasks (turning pages, preserving digitized archive copies of text, and the like).

In order to respond to this important recurring critique, we must explain more specifically what type of computational technology CASTR is: an *information processing* technology or, as it is defined within artificial intelligence, a "physical symbol manipulation" machine (Newell, 1982, p. 99). Nevertheless, it is a specific type of information technology.

Just as with any other technology, CASTR manipulates internal symbols (or *electronic signals*), each one encoding some object external to the system. As is said in philosophy: these signals stand for "something else," they "represent" it, and they are "proxies" for it. Even so, there is a difference between an ordinary information technology, such as a robot, and a CASTR type of technology. In the former, the signals stand for objects in the world. In the latter, the signals do not "stand for" such objects but for other symbols which are in a text. This means that CASTR is a *double layer symbolic* technology. Both layers are semiotic systems but they are different. Each one has its own syntax, semantic and pragmatic.

The first layer of symbols is internal to the machine itself, just as it is with any other *information processing technology*. They are, in fact, just electronic signals that are part of the data and programs of the machine. This layer of symbols defines a computer as such. The second layer of symbols pertains to what it is to be a textual semiotic system. If either of these two layers are left out, there is no CASTR technology. Erase the first layer and there is no real computer technology. Forget the second, there is no text.

In order to understand the CASTR technology, the two layers must not be confused. What the computer manipulates are the symbols of the first layer, not of the second layer. What the human reader manipulates are the symbols of the second layer. Only with these is the real *reading* realized. For these reasons, a CASTR technology cannot "read" a text (except in a metonymic meaning). Reading is reserved to humans; only they can read texts. For these same reasons it can only assist someone in reading a text[1], which is to assist him or her with interpretation of a text.

This dimension of *assistance* has to be underscored. A technology is a "tool" for reading, not a robot that reads texts. As a tool, it is a mediation or a medium by which the user *interacts* with the world or, as it has been underscored in artificial intelligence by Winograd and Flores (1986), the social science by Glaser (1967, 1998), and the digital humanities by McCarty (2005), it is to be seen as an assisting *manipulative action*.

The threefold theoretical grounding of CASTR technology

Defining CASTR as a symbolic technology requires a three-level type of explanation.

First, we have to explain the technology in terms of what a user can realize with such a tool: what are the cognitive tasks an expert reader intends to perform using this tool? Second, we must also explain it in terms of the various functionalities the tool itself can realize in order to accomplish the intended tasks. Finally, we can explain the tool in terms of the physical structure it must have to perform these cognitive tasks and functions.

By explaining a technology, these levels have to be distinguished because each one identifies different types of regularities, invariants, rules, and laws. We hope to show they are heuristic for understanding CASTR. We will, therefore, distinguish three levels of explanation: a cognitive, a functional, and a physical one.

The cognitive level

A first level of explanation pertains to the cognitive operations underlying reading. Compared to many other types of cognitive operations, such as perception or emotion that are applied to objects of some natural kind (apple, tiger, cloud, rock), or some internal psychological states (hunger, fear) or mental states (belief, desire), reading is applied to special objects that are not taken as themselves but for what they stand for. At its most basic meaning, reading is a cognitive operation applied to natural symbols and only to them.

And such a type of cognitive operation is not a simple one, mainly if the symbolic object is a text (Meunier, 1997). Indeed, a text cannot be reduced to its words or even to its sentences. It is an intricate structure of words and sentence. And researchers do not all agree on what exactly the structure of a text is; most scholars, nevertheless, agree that it is a travel into a semiotic structure:

> The important thing about the nature of a text is that, although when we write it down it looks as though it is made of words and sentences, it is really made of meanings. Of course, the meanings have to be expressed, or coded, in words and structures, just as these in turn have to be expressed over again - recoded, if you like - in sounds, or in written symbols. It has to be coded in something in order to be communicated; but as a thing in itself, a text is essentially a semantic unit. It is not something that can be defined as being just another kind of sentence, only bigger. (Halliday & Hassan, 1989, p. 10)

For our research purposes, we shall distinguish two interrelated reading strategies of these semiotic systems: a *micro* level and a *macro* level. Both systems are interpretative acts on symbols but they operate at different levels.

Basic reading

Within ordinary reading, we can distinguish two levels of linguistic cognitive operations. The first reading strategy is a micro, if not a local one. It pertains to the basic constituents (words and sentences) that have to be understood by the reader. It is often this level that we hope young children attain first so that they can read, with each word and sentence parsed and understood.

Yet real "reading" is not limited to this type of micro reading: it is more of a macro, if not a global, reading where one aims to understand some meta-organization of the text. As all text theoreticians will stress, a text is much more than a sequence of sentences; it contains some structural organization that a reader must identify and understand. For instance, a text contains many interrelated semiotic dimensions that participate in the construction of a theme, an argumentation, a narration, a description, a demonstration, a dialogue, and so on. Many theories can be found about the properties and structures of these textual constituents (Rastier, 2005; Adam, 1992).

Meunier, Jean-Guy, Poirier, Pierre, Danis, Jean, & Payette, Nicolas. (2012). Theoretical Grounding for Computer Assisted Scholarly Text Reading (CASTR). *Scholarly and Research Communication, 3*(2): 020116, 20 pp.

One important dimension of these theories is that these properties are not in themselves strictly grammatical phenomena, though they may present some strong regularities that belong to their kind (Rastier, 2001), rhetorical structure (Mann & Thompson, 1988) or logical structure (Hobbs, 1990).

Expert reading

Similar to basic reading, expert reading is a set of cognitive operations, only much more complex. In order to help us in our understanding of this expert reading, we shall use our two-level reading strategies – the micro and the macro – and seek to identify some of the more specific operations that are performed in expert text reading.

The expert micro-reading strategy

With a micro strategy, an expert reader must read a sentence and the words it contains in order to understand it. For instance, in reading the following much-translated and -paraphrased sentence from Kant's *Critique of Pure Reason,* an expert must first understand it at a micro-level:

> Our knowledge proceeded from two fundamental sources of mind: the first is in the power to receive representations. (n.p.)

He or she must understand each of the words and make sense of the whole sentence. But an expert reader will not be satisfied with this first-level reading: some finer and deeper meaning must be attained. There will then be a "digging" into the meaning; a variety of tools will be applied to the word and to the sentence[2] so as to unwrap what the sentence may really mean.

Some of these tools may focus on the linguistic properties of the word and sentence. For instance, a reader may dig into philological origins of the word searched. This may add some deeper understanding to the term under scrutiny. An example is from Darwin's *On the Origins of Species* (1980), where an expert reader may encounter the word "evolution" in the following sentence:

> *Every naturalist admits the great principle of evolution.*

An expert reader such as Robert Richards (1992) immediately relates this word to its philological original: the Latin word *evolvo*, which means *to unscroll a text*. This relation of the word to its philological origin will immediately orient Richards towards an interpretative thesis on Darwin.

Another tool may situate the word within the horizon of the linguistic dictionary of the text, the glossary, thesaurus, or the encyclopedia of a particular discipline. Other tools may relate each word to other texts of a personal, cultural, or historical relation to the sentence.

Many other examples of micro-reading could be given. In one sense, each strategy used by the expert reader aims to ground in some manner the meaning of the words and their role in a sentence. This allows the expert reader to unwrap some unforeseen meaningful dimensions that would otherwise be implicit at this micro-level.

Macro-reading strategies

Even if one can rigorously read a sentence and interpret each word, a micro-level reading is often insufficient to an expert reader. The real, deep meaning of the sentence is entangled with the rest of the text. It will often emerge after a parsing of many other sentences related to it in some manner or other. Some re-reading of the whole text may be necessary to understand the deep meaning of the use of a word in the local sentence. Hence a macro-reading is necessary.

An expert reader does not read sentences one after the other, extracting the meaning of each one. She or he aims at discovering some implicit meaningful *patterns or structure*s that underlie many sentences at once. This type of reading is often relevant to various types of understanding of the text. For instance, depending on the discipline that she is working in, the reader may be looking for a *thematic* organization, a *conceptual* structure, an *argumentation*, a *proof*, or a *narrative*. Hence, an expert reader will rapidly move from the basic-meaning micro-reading of a text up to its *macro-reading*, a reading level where meaningful latent structures are sought. Macro-reading can be seen as a type of *second-order reading*, for its role is one that classifies, categorizes textual objects, or situates parts in relation to the whole text.

In order to understand macro-reading, it will prove useful to distinguish two different *macro-reading* strategies: sequential and transversal. A macro-reading strategy is any organized set of consciously applied rules, the function of which is to make explicit tacit meaningful structure. With practice some, perhaps all, of these rules may become automatic.

In this type of macro-reading we can distinguish two main types of strategy: the sequential and the transversal.

The sequential strategy

The sequential strategy is one that most readers spontaneously move to when they realize macro-reading.

We define a sequential macro-reading as any set of rules applied while the reader is reading the text's sentences one after the other in order to extract one type or other of tacit meaningful structure. Although most macro-readers will not always apply a formally specified and studied macro-reading sequential strategy, slowly building their macro-reading strategy piece by piece, by trial and error, formal macro-reading strategies exist.

To illustrate the process of sequential macro-reading, we present a type of such reading, developed by Lacharité (1991), and that we shall call the thematic-pragmatic (TP) sequential macro-reading strategy. According to this strategy, all texts have a thematic and a pragmatic structure. Both structures can be organized as a tree to which can also be added, in some texts, a partial or complete logical structure, also organized as a tree. These hierarchical structures are not always self-evident. They are often hidden within the texts' basic and global meaning. The purpose of a macro-reading, according to the TP strategy, is precisely to uncover this hidden structure while one sequentially reads the text.

For instance, if we take the thematic structure of the text, it will take the form of a tree, the roots of which are the *theme* of the whole text. Each leaf is then seen as the

Meunier, Jean-Guy, Poirier, Pierre, Danis, Jean, & Payette, Nicolas. (2012). Theoretical Grounding for Computer Assisted Scholarly Text Reading (CASTR). *Scholarly and Research Communication, 3*(2): 020116, 20 pp.

subject of the text's individual sentences. The intermediary nodes will be the various subthemes that thematically unify the text (the simplest of which is the division of the text into sections, subsections and paragraphs, but it is worth mentioning this division rarely captures all about the relevant thematic structure).

The whole text is represented at each level of the tree: at the root, the whole text is represented by a single theme. At the first level, the whole text is represented by the themes of the first-level subdivision, and so on, up to the leaf-level where the whole text is represented by the subject of each sentence. One of the rules the strategy teaches is that there should be no holes at any level, no portion of text left without a theme. This normative rule is backed by the theory's belief that an author in control of her writing will intuitively (or perhaps purposefully) organize her text to fit this tree-structure. The student of the TP sequential macro-reading strategy must at first work hard to find a theme for every portion of text. Failure to find a theme for a portion either means that the macro-reader does not fully understand the text yet or, as a last resort interpretation, that the writer has produced a thematically sub-optimal text, either because she was not in full control of her means or because the text's thematic structure is in conflict with its pragmatic or logical structure. Of course, most thematic macro-readings of a text do not need to decompose the text's thematic structure all the way down to the subjects of its sentences.

These pragmatic, thematic, or even logical structures are but examples of this kind of sequential expert macro-reading. Many other structures exist. For instance, a literary critic may wish to discover some parallel narrative (a two-level reading) by extracting two interlinking structures pertaining to time, action, events, values, and so forth. A lawyer may, contrarily to pursuing the theoretical grounding and well formation of the text, pursue the contradictory structures in a testimonial verbatim. In a verbatim, a psychologist may wish to unveil the underlying hidden justifications, or "rationales," underlying apparently simple descriptions of behaviours. In analyzing a discourse, a sociologist may aim at revealing the true political motives entangled within a pretended "objective" explanation of an event.

Each one of these expert macro-readings depends on the disciplinary practices of the reader. No one is the dominant type. Developing models for these types of structures is an important research topic in many disciplines.

The transversal strategy

An expert does not always follow the natural course of a text, line by line, sentence by sentence, or chapter by chapter. An expert reader may often travel through the text in a transversal manner. He "reads" it as a whole so as to find in it some other structural types of links that are either internal to the text itself or external to its discourse (e.g., cultural, scientific, historical, etc.).

Some of links are internal to the text. For instance, in a well-structured text, one theme is normally linked to another one logically, narratively, or in some other way. Often these links between the themes do not appear serially through the various sentences. The node themes are often to be found at a distance from each other. Therefore, a

transversal reading is necessary. Such a reading will aim at organizing or unifying in some manner the internal thematic structure of a text.

Conceptual analysis is another type of such internal transversal reading. It will explore the various definitional and inferential properties of a concept that are often spread throughout the whole text. One will find here a definition, there a consequence, elsewhere an illustration, and so on. Unwrapping the conceptual structure will often rest on a rigorous transversal reading of the text.

Other links may be external. For instance, an expert reader may link part of the text to external information that modify or nuance the interpretation of the text which is the object of reading.

This information may originate in the personal background knowledge of the reader but may also be the cultural and scientific knowledge in which the reader lives and with which he will communicate his expert reading. It may even be dynamically produced through dialogue with known experts and commentators in the field.[3]

In fact, what happens here is that the interpreter puts the whole text in relation to a whole set of external representations that make more complex the semiotic system in which the reading takes place.

In this sense, expert reading brings into action a complex network of layers of symbolic structures, all of which can be called upon for an effective reading, depending on one's reading strategy. None of them are necessary, but many structures can co-operate so as to build a single interpretative path.

CASTR technology for assisting expert reading

Whether the reading is micro or macro level, it must always be viewed as a specific cognitive ability: the ability to manipulate symbols. In this sense, reading is a high-level semiotic ability that, in itself, unwraps into many subcognitive operations such as parsing, describing, memorizing, organizing, comparing, structuring, synthesizing, associating, abstracting, generalizing, and deducing ideas, themes, topics, concepts, theses, and so on. It is these many different cognitive abilities that a CASTR technology must assist, if not enhance, but surely not compete with.

Unfortunately, the related CASTR technologies have not always paid much attention to the various cognitive operations embedded in expert reading.

Jacques Virbel (1993), Stiegler (1994), and others were among the first to remind us of the complexity of the cognitive tasks underlying electronic text reading specifically. Vandendorpe has insisted on the implicit cognitive operations underlying the reading of a codex. Van Oostendorp and de Mul (1996) underscored the numerous cognitive aspects of electronic texts. McGann later reformulated this in terms of the revolution the electronic document has imposed on our reading and analysis.

Some researchers in the social sciences computer technologies have been slightly more sensitive to this cognitive dimension for assisting expert reading. For example, qualitative

Meunier, Jean-Guy, Poirier, Pierre, Danis, Jean, & Payette, Nicolas. (2012). Theoretical Grounding for Computer Assisted Scholarly Text Reading (CASTR). *Scholarly and Research Communication, 3*(2): 020116, 20 pp.

analysis (Glaser & Strauss, 1967) with computer applications such as NVivo (NUDE*IST), Atlas.ti, HyperRESEARCH QDA Miner (Barry, 2009; Lewis, & Maas, 2007).

Today, Lexist is one of the rare tools that directly assists expert reading of e-texts. Unfortunately, its scope is limited to ancient texts. More recent projects in the USA and Canada, such as WordHoard, MONK, and TaPOR (Unsworth, 2000) offer more integrated environments for scholarly reading of electronic text. Pliny is another recent technology specially dedicated to reading assistance (Bradley, 2007). Siemens' research program (see Siemens, Willinsky, Blake, Newton, Armstrong, & Colahan, 2008) aims at better understanding reading in the digital age. More and more formal content annotation technologies also aim at assisting some aspect of expert reading (Cieri & Bird, 2001; Bird & Liberman, 2001).

If a reading technology is really to assist in the process of not just basic reading but expert reading, a better understanding of these operations is required. This will allow the development of applications that correspond to the reading operations performed by expert readers. And with the growing wave (if not tsunami) of electronic books, CASTR will have to be developed to a finer capacity to assist this expert reading of text.

The functional level

If it is essential to examine the cognitive level to understand a technology, it is also necessary to understand how these cognitive operations are translated into functions that are implemented by a computational technology: possible algorithms. In other words, a technology also must be explained in what Dennett (1987) calls a "design stance," or what Pylyshyn (1984) and Newell (1982) both call a *functional level*. This is the object of the second level of explanation.

As repeated in the literature, reading is in the hands of the reader. It is always the reader who ultimately interprets the text. In this view, the computer is but a tool in the process. And for an expert reader the technology must be viewed as an assistant that "blindly" applies functions to a text, the role of which it is to manipulate the "proxies" of the symbols written in the original text.

For example, only an expert reader can interpret Darwin's *On The Origin of Species,* as defending a dynamic theory of *evolution*. No computer could be so sophisticated so as to arrive at such a conclusion. But a computer can offer some functions that assist this expert reader in manipulating the "proxies" of the original text in order to help him discover and defend such a theoretical interpretation. This can be realized, for instance, by a simple computer function: RECALL all the sentences that use the word *evolution* in the text, produce the HISTORICAL histogram of the introduction of this word through the 53 editions of *The Origin of Species* etc. More sophisticated functions are possible, and we shall present some below. Each of these functions does not interpret the text as such. In the end, the choice of these functions is always decided by the expert reader.

In order to assist the myriad of cognitive operations an expert reader accomplishes in the reading process, an appropriate CASTR technology must offer many such functions and combinations of functions.

Here we offer a classification hypothesis for the variety of these functions. For the purpose of research organization, we shall here distinguish between three main classes of functions. The first class contains *transformation* functions: these functions are applied to the original text symbols and transform the text into a different one. The second class contains *meta*-analysis *function*s that are applied not to the original text but to the transformed text. Their role is mainly to assist the analysis of the results of the application of a CASTR reading to the text. Finally, a third set allows the management of the whole set of transformation and analysis functions.

Transformation functions

When reading a text, there are a myriad of basic material tasks that are realized by an expert. Many of them will not even be noticed, because they have become integrated in the concrete reading practices, but they all transform the text in one manner or another. Some of them enhance the text, others reduce it.

An example of the first type is highlighting a sentence with a yellow marker. Underlining with a pencil is, formally speaking, an enhancing transformation of a text. These both add symbolic marks (a yellow overlapping mark, a pencil trace) that have special meaning for the reader.

Other tasks can be reductive. For instance, for whatever reason a reader may decide to pick up a subset of passages from the original text. One may delete sentences from the original text (e.g., eliminating the title of each page, the footnotes, or some paragraphs). Another, more sophisticated reductive operation is the production of a subset of particular sentences that contain a key word. A *concordance* is, in one sense, a reductive transformation operation: The whole corpus is now reduced to the set of these sentences.

There exist more complex and subtle transformation functions. For instance, when reading Aristotle, an expert reader may have to underline what he thinks is the subject of a particular sentence according to his knowledge of ancient Greek. He may mark this by circling the word, adding an arrow directed to the verb of the sentence, and adding the symbol (SUBJECT). In other words, a basic syntactic analysis is applied and a morphological tag is added. Some other functions are even more sophisticated. For instance, an expert reader may decompose an argument, gloss over it, and add a comment to a passage.

What all these simple or sophisticated functions have in common is that they have transformed the text in one manner or another. Some of them enhance the text with new symbols; others reduce it to a more simple form. They all have taken for input a set of symbols and have either added to it some new symbol or deleted from it some symbols. Here are examples of such functions.

Pre-processing functions

Cyberworld texts do not come in nicely edited e-books. Many have been digitized only with concern for their storage and retrieval. Expert reading has not experienced primary consideration in the area of design. Such digitized *corpora* do not always come with shared editorial standards (for instance a Text Encoding Initiative [TEI]). And often many simple operations have to be applied to the text so as to make it readable.

Meunier, Jean-Guy, Poirier, Pierre, Danis, Jean, & Payette, Nicolas. (2012). Theoretical Grounding for Computer Assisted Scholarly Text Reading (CASTR). *Scholarly and Research Communication, 3*(2): 020116, 20 pp.

Even a basic query in an e-text library may produce a huge amount of documents often wrapped up in various meta-codes and meta-tags, sometimes themselves wrapped up with linguistic tags (some morphological, some syntactic, and even some basic-semantic). For instance, if a scholar wishes to read from the Internet a set of articles on a particular subject, he will have to abstract and elide many unwanted peri-texts (publicity, URL, etc.) in order to produce significant data for his reading process. This significant data is not always related to linear texts (some readers may only want to concentrate on graphs, images, even specific handwriting in the case of paleography).

So, even before reading begins, a CASTR technology may have to offer a set of "pre-processing functions" such as CLEANING a text so to transform its "non standard format" into a compatible one. After this, a FILTERING function may be applied to render the text susceptible to an expert reading.

Browsing and viewing functions

Once a text is constituted, many classical browsing functions have to be offered, such as FIND and RETRIEVE functions. Other functions allow different types of VIEWING (zoom, rotate, divide pages, segment, compare versions, and so on). Although many of these functions are simple, they are essential when a text takes the form of an e-book. What was integrated in the codex form has to become evident in the digital forms. A typical set of such reading tools is found in PDF formatted texts (see for instance Adobe Reader).

Marking functions

A third set of functions assists the attentive reading practices an expert often applies to his text: practices that he or she has developed through the years and that are often idiosyncratic to the reader.

We can group them under MARKING functions that are applied on a text. For instance, HIGHLIGHTING, UNDERLINING, INDENTING, etc. All these marks add non-alphanumerical symbols to a text. Many of these functions are offered by basic reading tools.

Annotative categorizing function

An expert reader will often classify or categorize some part of the text in some manner, be it at the level of the word, the sentence, or larger parts such as paragraphs, chapters, etc. This is often the core of expert reading. Formally speaking, these operations are functions that add meta-symbols to the original text symbols. A simple example is:

> LOVE THY NEIGHBOUR

that could receive a host of meta-symbols such as:

> (((LOVE (verb) THY (poss. article) NEIGHBOUR (name)) Command) Religious), etc.

Many technical terms have been used to name this type of transformation function:

- In classical digital humanities: TAGGING or ENCODING functions (e.g., the editorial formatting of the Text Encoding Initiative, or TEI);

- In computational linguistics: ANNOTATION or MARK UP functions (Chiarcos, Dipper, Götze, Leser, Lüdeling, Ritz, & Stede, 2008) (e.g., the morphological, syntactical, semantic, pragmatic, rhetorical discourse annotation scheme); and
- In Artificial Intelligence and psychology: CATEGORIZING (Harnad, 1990).

A CASTR annotation categorizing function must offer not only a set of categories but more so the means to easily produce any sort of such category or annotation. Each such type of category must be distinct but also be also related to each other so as to allow a structuring of some sort (e.g., hierarchy). This type of function will often be the core challenge of an adequate CASTR technology.

Commenting functions

An expert will often instantiate his own interpretation process in producing a variety of commentaries on a passage. As considered by the Grounded Theory methodology (Glaser, 1998; Glaser & Strauss, 1967), COMMENTING is often part of the interpretative creativity that accompanies a text interpretation. It is often done without rules of writing, grammar, or style. These commenting functions are essential and have to be "harmonized" as much as possible with the reading process. A very common type of such a commenting function takes the form of a *paraphrase* of the original text. Another one is a strict commentary such as a critique, a specific analysis, a citation, or even a translation. All these can be thought of as peri-text.

CASTR commenting function must here offer the possibility to easily produce any sort of such types of comments. It must also allow the user to categorize these comments in any manner so that they are not mixed one with the other. Finally, these comments must be easily manageable.

Hypertextual functions

One constant operation an expert reader will perform in either a local or transversal reading of the text is to relate it to some other texts. These relations can be internal to the text itself, for instance, relating one sentence to another with one being a premise and the other the conclusion. These types of links are often the ones that translate, in functional terms, the various cognitive operations at work in a pragmatic and logical analysis of text, as we have presented above.

Some other relations are external. They relate part of the text to other texts in the technical literature. For instance, a sentence can be related to its own translations, or to the direct comment another author may have expressed about it.

In technological terms, these last types of relations are called hyperlinks. They create an internal or external *hypertext* that are often created by the reader himself. Increasingly, we see small robots or agents that dive in the cyber-textual world to find "something similar" to what is under scrutiny, as it is called. What a CASTR technology must offer is a set of functions that assist the creation, discovery, and organization of such hyperlinks. For instance, they may support the organization by offering various types of trees and graphs for representing the argumentative structure or the "web" of hyperlinks discovered.

Meunier, Jean-Guy, Poirier, Pierre, Danis, Jean, & Payette, Nicolas. (2012). Theoretical Grounding for Computer Assisted Scholarly Text Reading (CASTR). *Scholarly and Research Communication, 3*(2): 020116, 20 pp.

Meta-analysis function

Once a text has been read in an expert manner, or maybe during the reading itself, a new transformed text has been produced. Some part of this new text is the digitized text itself, but other parts will have all the added tags, categories, and so on. Yet other parts will contain comments. Hence, the original digitized text has now become a multilayered text. In a true expert reading, an analysis of this multilayer text is bound to happen. The analysis aims at organizing, in some manner, the various dimensions of what the reading has produced. This analysis constitutes another step in the interpretation of the text; it is specific and should be distinguished from the sophisticated complex text analysis one finds in text mining strategies. Here the analysis pertains strictly to the results the various transformation functions have produced. They are, in one sense, meta-analysis functions applied to the result of transformation functions as such.

Using this function, a reader may for instance want to *summarize* or *synthesize* his findings; more often he may want to *organize* them. For instance, he may wish to recover the various paraphrases, comments, and hyperlinks produced, but he may also wish only to organize the paraphrases of the text such that, when well done, a personal summary of his understanding of the text is produced. To this, he may wish to link each part to the various types of commentary he has added to the text and even his own paraphrases.

Other types of such a function can be statistical, allowing the reader to qualify numerically some of his findings, or categories of his findings.

Managing functions

Finally, an adequate CASTR technology must offer a variety of meta-functions, which allow the management of the preceding functions. These functions are often similar to the ones found in database management systems.

Flexibility and modularity of the transformation functions

Many of the preceding reading functions concentrate on various levels and can be called upon simultaneously or sequentially. For instance, through the reading process, a segment may be tagged as being of a "thematic" category. At the same time, it may be tagged as a *pragmatic assertion*. Another segment could reveal its *morphological features* while being seen as a *lexical definition*. These processes can be seen by the expert reader as various "jumps" between reading levels, categories (tags), semiotic registers, and so on. Even if these "jumps" seem almost chaotic through the reading process, they progressively bring about the complex organization of the data through expert reading. They are intimately close to the creative process underlying the interpretation of text. Such an organization process implies constant back and forth movement between the many diverse micro- and macro-textual reading operations.

Because these functions are often applied simultaneously and in a variety of ways, an adequate CASTR technology must offer a degree of flexibility and modularity. For instance, it must allow constant exploration of inter-relationships between all these functions, such as marking, annotations, and commenting. It may also offer support producing some qualitative (e.g., queries that merge different annotations levels),

quantitative (e.g., statistics on certain lexicometrical dimensions of the text), or even structural functions (e.g., text classification, machine learning deductive algorithms).

The physical level: The design of a CASTR technology

The last level of explanatory architecture of a digital technology relates to its physical support, or digital implementation. This level explains how the cognitive tasks and functions are realized by a specific material technology. To say it more concretely, these functions must be physically implemented in a positive/negative electronic digital flow machine called a computer. For CASTR, this means two important things: 1) The original text or set of written symbols now have a new physical support: they are encoded in a second layer of e-symbols; the text has become an electronic document or an e-book or a sequence of e-symbols[4], which are now the "data" to be dealt with; 2) The functions and operations that manipulate these e-symbols are themselves constrained by a specific computer technology (for example, on a laptop, on a server, in an XML format) and they must be manipulated in some sort of structure of relational database. This applies to the entire electronic document technology, which is a computer domain in itself. A CASTR technology can be viewed as an add-on to this technology.

In designing a computer technology to support an expert reading, one often takes a top-down approach: an e-document exists and an expert reader must try to accommodate it. In other words, a tool exists and the expert must find the object that fits to it (e.g., nails for a hammer). In the textual world, this type of approach may be useful in a marketing strategy but may fail if one expects its adoption by expert readers. A good example is that selling the actual e-book technology may well work for general text reading but its adoption by scholars is bound to find high resistance, except for cursory reading, because of its limited functionalities that are not yet sensitive to the needs of expert reading.

From our perspective, the design philosophy must be the opposite: a CASTR technology must first answer to cognitive requirements and functional operations of expert reading. Unfortunately, finding this compliant technology is in itself a research program, so one must often proceed by trial and error.

As above, the real requirements for an expert, while he or she is reading, are classification and categorization. A rich CASTR technology must allow rich annotations, commentaries, strategies, and so on, and should prove flexible enough to accommodate different styles of expert reading, both at the micro and macro levels. For instance, it should allow for the application of multiple simultaneous strategies, such as different annotations on a same corpus and cross-referencing between these different annotations sets. Another important design goal for the system is to make sure that it can interface with other modules, either before, after, or during the annotation process. Finally, it should also prove to be extensible so that users can design their own add-ons – task specific components that can be integrated into the system and provide functionalities for either individual users or a whole community of experts.

We briefly present here a concrete CASTR platform, after a sample of the technology that is built in the spirit of the preceding theoretical foundations[5]. At the technological level, our CASTR system is a Java desktop application. Despite some widely acknowledged weaknesses of the language itself (verbosity, lack of closures, awkward

generics, and so on), Java remains the *de facto* standard in today's programming world, providing significant advantages like portability between different operating systems and a wide selection of third party tools and libraries.

This CASTR system uses XML files for input formats and a relational database for permanent storage and dynamic manipulation of the data, hence leveraging the strengths of both these storage technologies. Its platform has two main components: a parser for converting XML data files into a hierarchical data structure suitable for annotation and the graphic user interface used for the annotation process itself. We will describe each of these in turn.

Different corpuses may have very different structures, with different granularities. An expert reader may want to annotate a book divided in chapters, which are in turn divided in sections that are divided in paragraphs. Or she may want to use a similar structure, but also work at the individual sentence level. Another reader might work on multiple classes of text segments produced by some clustering algorithm previously applied to them. In order to accommodate these different needs, an XML parser is available. It makes very few assumptions about the structure of the input document. Basically, the only constraint is that the document has to be tagged with some well-formed XML, whether this tagging is done manually or by some pre-processing tool. Names and attributes of each XML element are read by the parser and a tree structure is produced and stored in a relational database. Each node of the tree is stored with the text contained within the XML tag. If we go back to the book example, a <BOOK> element could be the root node of the tree, followed by <CHAPTER> nodes, containing <SECTION> nodes, and so on. For the clustered corpus example, we could have a root <CORPUS> node, with <CLUSTER> and nodes as children. The storing of XML attributes provides the user with the means for keeping track of different information about a text and for maintaining links with other systems (e.g.,).

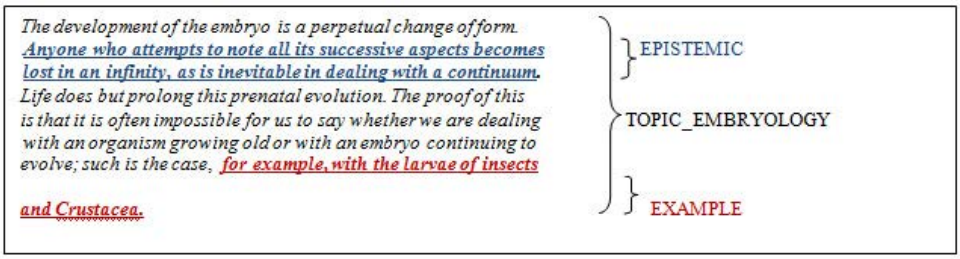
The document tree resulting from this parsing process is then represented in the user interface as a graphical tree in which the user can navigate, much like a hierarchical file system on most of today's computers. As the user clicks on the tree nodes, the text content associated with that node is displayed in a column to the right of the tree. All of the annotation process takes place within the context of that tree.

To put it in the simplest possible way, an annotation is a label that you can apply to a node. Yet, richer and more general annotations are possible. For instance, these annotations can have attributes that are to be filled in by the user. For example: One categorizes a node as being a *COMMENT* and afterwards fills in the *text* attribute, that is in fact the comment on the sentence, paragraph, or whatever level of text organization the node refers to. A simple comment is a fairly trivial example, but the system allows for an extensive library of labels that can be used in annotations. These labels come from a library that is also hierarchically organized. Though we will not go into technical details here, the label library is stored as an XML file that the user can modify easily. Different sets of labels used in different annotation strategies can

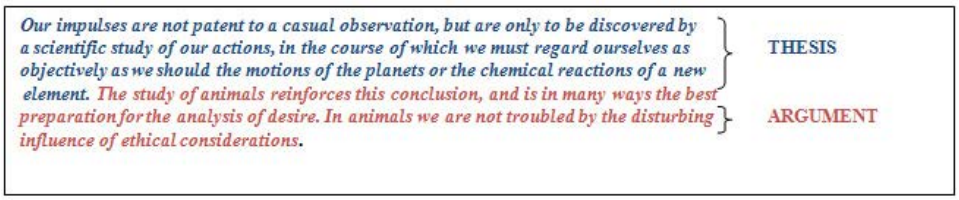
Meunier, Jean-Guy, Poirier, Pierre, Danis, Jean, & Payette, Nicolas. (2012). Theoretical Grounding for Computer Assisted Scholarly Text Reading (CASTR). *Scholarly and Research Communication, 3*(2): 020116, 20 pp.

co-exist side by side in this library, each forming a different branch of the whole label tree. Since annotations can be assigned to a single node, a given paragraph could be annotated with both a *STRATEGY_A / SUBDIVISION_1 / LABEL_X* label and a *STRATEGY_B / SUBDIVISION_2 / LABEL_Y* label.

Example 1



Example 2



The label library is also presented to the user as a graphical tree, shown on the right-end side of the application window. Creating an annotation is a matter of dragging a node from the label library and dropping it on a node from the document tree. An annotation form then appears to the right of the text column and the user can fill in the required attributes. Among the attributes of an annotation, there can be links to nodes other than the one that the annotation is associated with. This allows the reader to record relationships amongst different parts of a text. For example, if segment twelve is a refutation of segment seven, it can be annotated with the *REFUTATION* label (to be found in the appropriate branch of the label library) and the *refuted_segment* attribute of the annotation filled with the number seven. Using a greater number of these linking attributes, one can express more complex relationships, such as one segment stating an opposition between two others. Ultimately, these relationships allow to create post-reading paths through the data and allow a timeline different from the initial sequential text. For example, the reader could create a reading path like:

> THESIS → REFUTATION → ARGUMENTS *(rhetoric sequence)*
> **(seg.3/seg. 8) (seg. 12/seg. 4) (seg. 33/seg. 10)**

He may then want to join previous annotations or commentaries to that path to eventually nuance the interpretation as, for example, the combination of:

> THESIS → REFUTATION → ARGUMENTS *(rhetoric sequence)*
> **(seg.3/seg. 8) (seg. 12/seg. 4) (seg. 33/seg. 10)**

He may then want to join previous annotations or commentaries to that path to eventually nuance the interpretation as for example, the combination of the THEMATIC dimension:

> THESIS → REFUTATION → ARGUMENTS *(rhetoric sequence)*
> **(seg.3/seg. 8) (seg. 12/seg. 24) (seg. 33/seg. 10/seg. 134)**
>
> EMBRYOLOGY INSECT/CRUSTACEA *(thematic)*

In this example, relationships have been at the same time established between multi-layer categories and different parts of text to illustrate that the arguments the author deploys in refuting a thesis are related to a specific thematic, here "insects and crustaceans." This reading path could eventually be compared to other sources, could be modified, and so on. What is important is that these functions allow one to go further than simply attaching codes to segments: they allow the establishment of a strong internal structure in the primary source, as for the categories and the transformed texts that emerge from it. A new subtext is created here (structured by segments 3, 8, 4, 33, etc.). Different reading paths and transformation functions (commenting, lexical extracting, and so on) could thus emerge from it.

During the course of the annotation process, an expert reader can also realize that a finer grain of annotations is needed to uncover the complexities of the text at hand. Hence, the system allows for the modification and extension of the label hierarchy while the annotating is taking place. For example, a reader trying to identify the different themes contained in a text could quickly add new themes and subthemes under the *THEME* branch of his label hierarchy. Once the annotation process is complete (or perhaps during the process) the relational database backing the application can be queried for statistical information about annotations applied to the text. These queries can be written using the SQL structured query language, which is much easier to grasp for the average user than the different XML query facilities. Once all these annotations have been added to the text, various sorts of management, recall, and quantitative analysis function can easily be applied to these. For instance, a reader could search for all the segments tagged as *DEFINITION of X*.

Conclusion

A CASTR technology is an add-on to this electronic document technology, one that will be requested more and more as the e-book gains academic and professional readers. In order to assist adequately this expert reading, it has to offer modularity and flexibility. In order to be able to offer this modularity and flexibility, the technology has to go beyond the sole "code and retrieve" paradigm that is often related to qualitative research. If code and retrieve are among the principal functions that allow a constant structuring of data by interrelating the macro- and the micro-textual dimensions, these functions, when used in wider expert reading processes, have to be combined with other functions. Some studies explore the merging of these numerous functions with the code and retrieve processes. Some have given rise to software that interrelate the reading process with qualitative and quantitative functions (e.g., QDA miner). Code and retrieve are done

Meunier, Jean-Guy, Poirier, Pierre, Danis, Jean, & Payette, Nicolas. (2012). Theoretical Grounding for Computer Assisted Scholarly Text Reading (CASTR). *Scholarly and Research Communication, 3*(2): 020116, 20 pp.

here along with the merging of statistical and visualization tools. Other researchers have instead concentrated on interfaces allowing a complex visual structuring of the data that is easily readable and modifiable through the interpretation process.

Working on the platform presented here, we also aim at interrelating numerous functions to the code and retrieve processes, this time by developing and exploring an environment that allows the combination of different annotation strategies and different structuring processes. More specifically, we aim at integrating these processes to the different transformation functions presented above. Since these functions require numerous functional operations, we aim at developing as much as possible a platform that is flexible enough to satisfy these requirements.

Notes

1. Such a question is not original and can be asked of many technologies: a pen does not write a text! But can it assist a writer in the manipulation of written symbols? The same question can be asked about a magnifying lens: it does not read a text, nor do printers or scanners. They only assist the manipulation of the carriers of the textual symbols. The human person is in the driver's seat. To be a technology that assists expert reading, CASTR must ultimately assist the manipulation of this second layer of structured and interpretable sets of symbols.

2. They vary from discipline to discipline.

3. New types of highly intellectual social networks are emerging. The knowledge produced through reading is shared, in a specific academic community, through a technological social network and therefore influences the readers directly.

4. One amazing but unforeseen dimension of this transformation of the paper text into electronic form is that it has allowed not only the insertion of textual symbols but also audio and visual symbols. What were traditionally three distinct semiotic forms find themselves on the same physical support or medium. This changes radically the nature of the "textual" corpus itself. Now a "corpus" may include linked texts, images, sounds, and even animations. It will not be surprising to find on the same digital support an English sentence expressed in a string of text, a sound file, and perhaps even a video clip, all of which make a coherent contribution to the meaning of the whole. This new type of "multimedia text," precisely because of this change of physical support, opens up new technological challenges for reading and analysis.

5. This CASTR technology is part of ongoing research projects on Computer Assisted Reading and Analysis of Text (CARAT) of the Laboratoire d'analyse cognitive de l'information (LANCI) at the Université du Québec à Montréal.

References

Adam, Jean-Michel. (1992). *Les textes: types et prototypes*. Paris, FR: Nathan.

Bird, Steven, & Liberman, Mark. (2001). A formal framework for linguistic annotation. *Speech Communication, 33*(1-2), 23-60.

Bradley, John. (2007). Thinking differently about thinking: Pliny and scholarship in the humanities. *Literary and Linguistic Computing, 23*(3), 263-279.

Barry, Christine A. (2009). Choosing qualitative data analysis software: Atlas/ti and Nudist compared. *Sociological Research Online, 3*(3). URL: http://www.socresonline.org.uk/socresonline/3/3/4.htm .

Chiarcos, Christian, Dipper, Stefanie, Götze, Michael, Leser, Ulf, Lüdeling, Anke, Ritz, Julia, & Stede, Manfred. (2008). A flexible framework for integrating annotations from different tools and tag sets. *Traitement Automatique des Langues, 49*(2), 271-293.

Cieri, Christopher, & Bird, Steven. (2001). Annotation graphs and servers and multi-modal resources: Infrastructure for interdisciplinary education, research and development. *Proceedings of the ACL 2001 Workshop on Sharing Tools and Resources, 15*, 23-30.

Darwin, Charles. (2008). *On the origin of species*. From the 1872 edition. N.P.: Mobi Classics.

Dennett, Daniel C. (1987). *The intentional stance*. Cambridge, MA: MIT.

Glaser, Barney G., & Strauss, Anselm L. (1967). *The discovery of grounded theory: Strategies for Qualitative Research*. Chicago, IL: Adline.

Glaser, Barney G. (1998). *Doing grounded theory - Issues and discussions*. Mill Valley, CA: Sociology Press.

Habermas, J. (1984). *Theory of communicative action. Vols. 1-2.* Boston, MA: Beacon Press.

Halliday, Michael Alexander Kirkwood, & Hassan, Ruqaiya. (1989). *Language, context, and text: Aspects of language in a social-semiotic perspective*. Oxford, UK: Oxford University Press.

Harnad, Steven . (1990). The symbol grounding problem. *Physica, D 42,* 335-346. URL: http://www.digitalhumanities.org/dh2007/abstracts/xhtml.xq?id=124 .

Hobbs, Jerry K. (1990). *Literature and cognition.* Stanford, CA: Center for the Study of Language and Information.

Lacharité, Normand, Poirer, Pierre, & Meunier, Jean-Guy Meunier. (1991). Sémantisme et Représentation. *Systèmes et Cognition, no. Y6.* [Montréal] : Dép. de philosophie (UQAM).

Lewis, R. Barry, & Maas, Steven M. (2007). QDA Miner 2.0: Mixed-model qualitative data analysis software. *Field Methods, 19*(1), 87-108.

Mann, William C., & Thompson, Sandra A. (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text, 8*(3), 243-281.

McCarty, Willard. (2005). *Humanities Computing.* London, UK: Palgrave.

Meunier, Jean-Guy. (1997). La Lecture et l'Analyse de Textes Assistées par Ordinateur (LATAO) comme système de traitement d'information. *Sciences Cognitives, 22,* 211-223.

Meunier, Jean-Guy. (2009). Theoretical Grounding for Computer Assisted Expert Text Reading (CASTR). *New Knowledge Environments, 1*(1). URL: journals.uvic.ca/index.php/INKE/article/view/161/175 [September, 2011].

Newell, Allen. (1982). Intellectual issues in the history of Artificial Intelligence. In F. Machlup and U. Manfield (Eds.), *The Study of Information: Interdisciplinary Messages* (pp.187-228). New York, NY: Wiley.

Pylyshyn, Zenon W. (1984). *Computation and Cognition. Towards a foundation for cognitive science.* Cambridge, MA: MIT.

Rastier, François. (2005). Pour une sémantique des textes théoriques. *Revue de sémantique et de pragmatique, 17,* 151-180.

Rastier, François. (2001). *Arts et sciences du texte*. Paris: PU de France.

Richards, Robert J. (1992). *The meaning of evolution: The morphological construction and idological reconstruction of Darwin's theory.* Chicago, IL: University of Chicago Press.

Siemens, Ray, Willinsky, John, Blake, Analisa, Newton, Greg, Armstrong, Karen, & Colahan, Lindsay. (2008). A Study of Professional Reading Tools for Humanists. URL: http://etcl-dev.uvic.ca/public/pkp_report/ [September, 2011].

Stiegler, Bernard. (1994). Machines à écrire et matières à penser. *Genesis, 5*, 25-49.

Unsworth, John. (2000). Scholarly primitives: What methods do humanities researchers have in common, and how might our tools reflect this? *Symposium on Humanities Computing: Formal*

methods, experimental practice. London, UK: King's College.

van Oostendorp, Herre, & de Mul, Sjaak. (Eds.). (1996). Cognitive aspects of electronic text processing. *Advances in discourse processes*. Norwood, NJ: Ablex.

Virbel, Jacques. (1993). Reading and Managing Texts on the Bibliothèque de France Station. In P. Delany and P. Landow (Eds.), *The Digital Word: Text Based Computing in the Humanities* (pp. 31-51). Cambridge, MA: MIT.

Winograd, Terry, & Flores, Fernando. (1986). *Understanding computers and cognition*. Norwood. NJ: Ablex.