# UC Irvine
## Software / Platform Studies

### Title
System Intentionality and the Artificial Intelligence Hermeneutic Network: the Role of Intentional Vocabulary

### Permalink
https://escholarship.org/uc/item/3rd2s695

### Authors
Jichen, Zhu
Harrell, D. Fox

### Publication Date
2009-12-12

Peer reviewed

eScholarship.org Powered by the California Digital Library
University of California

# System Intentionality and the Artificial Intelligence Hermeneutic Network: the Role of Intentional Vocabulary

Jichen Zhu*
Georgia Institute of Technology
686 Cherry Street, suite 336
Atlanta, GA, USA
jichen@gatech.edu

D. Fox Harrell
Georgia Institute of Technology
686 Cherry Street, suite 336
Atlanta, GA, USA
fox.harrell@lcc.gatech.edu

## ABSTRACT

Computer systems that are designed explicitly to exhibit intentionality embody a phenomenon of increasing cultural importance. In typical discourse about artificial intelligence (AI) systems, system intentionality is often seen as a technical and ontological property of a program, resulting from its underlying algorithms and knowledge engineering. Influenced by hermeneutic approaches to text analysis and drawing from the areas of actor-network theory and philosophy of mind, this paper proposes a humanistic framework for analysis of AI systems stating that system intentionality is *narrated* and *interpreted* by its human creators and users. We pay special attention to the discursive strategies embedded in source code and technical literature of software systems that include such narration and interpretation. Finally, we demonstrate the utility of our theory with a close reading of an AI system, Hofstadter and Mitchell's *Copycat*.

## Keywords

Software studies, Artificial intelligence, Hermeneutics, Critical Code Studies, System intentionality

## 1. INTRODUCTION

Human interaction with technical artifacts is often mediated by treating them as if they were alive. We exclaim "my car doesn't *want* to start," or "my computer *loves* to crash." Yet, of increasing cultural importance are computer systems designed explicitly to appear intentional. Compared with more instrumental programs, such as *Adobe Photoshop*, these intentional systems seem to produce output *about* and *directed at* certain things in the world rather than the mere execution of algorithmic rules. These systems, often produced in artificial intelligence (AI) practices and increasingly in digital media and electronic literature, exhibit complex behaviors usually seen as the territory of intentional human phenomena, such as planning, learning, creating and carrying on conversations. More importantly, they seem to display beliefs, desires and other mental states of their own.

Intentional systems are of particular relevance to various areas in digital arts and culture not simply because the latter provides a vibrant environment for experimentation with meaning, interaction, and context. Indeed, many salient examples of intentional systems are from music (e.g., George Lewis's interactive music system *Voyager*, Gil Weinberg & Scott Driscoll's robotic drummer *Haile*), visual arts (e.g., Harold Cohen's painting program *AARON*), and storytelling (e.g., Michael Mateas's drama manager in *Façade*). The relatively unexplored phenomenon of system intentionality also provides artists and theorists with novel expressive possibilities [26].

To fully explore the design space of intentional systems, whether in the forms of art installations, consumer products, or otherwise, requires a thorough understanding of how system intentionality is formed. Many technologists regard the phenomenon of system intentionality as a technical property of a system, directly proportionally to the complexity of the algorithm and knowledge engineering process, whereas humanists will be quick to point out its contingencies on cultural and social settings.

The discussion of intentionality in the context of computing can be traced back to the series of debates concerning AI among scholars from different fields during the 1980s and the early 1990s. Among the various approaches towards system intentionality, Daniel Dennett [6, 7] proposed the core ideas of *intentional systems theory*. According to Dennett, one of the most important strategies that people use to predict the behaviors of humans, animals, artifacts, and even themselves is the *intentional stance*. It requires treating those entities as rational agents with beliefs and desires in order to predict their potential behaviors [5]. He subsequently defined the systems to which we apply intentional stance as intentional systems [6]. For instance, we may not know exactly how a Roomba autonomous robotic vacuum cleaner is designed or constructed to traverse a room, but we can nevertheless make sense of and predict its behaviors to a certain degree by formulating our interpretations of its beliefs and desires. In other words, the apparent system intentionality is attributed by the users of these systems.

However, at least one issue remains. In his dissertation, Seel [23] has shown that we can apply the intentional stance to almost all artifacts that we interact with. This observation raises the question about the boundary of Dennett's definition of intentional systems. To the majority of us, certain digital artifacts afford intentional readings more easily than others. A robotic drummer, for instance, supports intentional readings more readily than a "hello world" program. This paper aligns with the core of Denett's theory — the intentionality of the digital systems under study is *not* their

*Current address: Department of Digital Media, University of Central Florida, 12461 Research Parkway, Orlando, FL, USA

© Digital Arts and Culture, 2009.
Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission from the author.
Digital Arts and Culture, December 12–15, 2009, Irvine, California, USA.

intrinsic technical property, as many computer scientists and theorists may believe. Meanwhile, it is also important to further develop Dennett's theory so that we also take into account the active role of system authors in the formation of system intentionality. This phenomenon is not only a result of human users' evolutionary skills, but also as a product of system authors' technical and discursive practices.

In this paper, we propose the *AI hermeneutic network*, a new framework to highlight authors' narration of system intentionality as well as users' interpretation [25]. More specifically, we call critical attention to the use of intentional vocabulary as a key component of the author's discursive strategies in their narrations. Section 2 first draws the paper's theoretical framework from relevant theories from both the humanities and the AI community, including Hayles's work on A-life researchers' discursive strategies and McDermott's and Agre's observations of intentional vocabulary in the AI practice. Next, Section 3 introduces our new construct of the *AI hermeneutic network*, which argues that system intentionality arises from a complex meaning-making network that incorporates software authors' discursive narration and users' hermeneutic interpretation of system intentionality in a board social context. Finally, Section 4 demonstrates the effect of our AI hermeneutic network through a close reading of a real AI system, *Copycat*. In our analysis, close attention is paid to the authors' use of intentional vocabulary in their narration of system intentionality. In addition to the source code of *Copycat*, we look closely into a substantial corpus of the technical literature produced by the system authors, which is a rich and yet relatively unexplored areas in software studies and critical code studies [18].

# 2. THEORETICAL FRAMEWORK

The topic of intentionality is of longstanding concern in philosophy. In the context of AI, it is commonly understood as "aboutness" [1, 8, 24] or defined as "that property of many mental states and events by which they are directed at or about or of objects and states of affairs in the world" [22]. Reintroduced by Franz Brentano [4] in the late nineteenth century and taken up by Husserl [16], the concept is considered as the linkage between the "inexistence" of human mental phenomena and the material establishments and states in the world. Intentional mental states, which include beliefs, desires and other states, are not free-floating thoughts, but are always *about* or *directed at* something. This means that we do not just have beliefs and desires in their abstract forms. Instead, they are *always* about certain states (e.g., I believe that it is going to rain tomorrow) or directed at certain objects (e.g., his desire for a sports car).

Many scholars have insisted that intentionality is an ontological property of the privileged human existence and therefore is not applicable to machines. One of the most renowned examples is John Searle's Chinese Room argument [22]. In comparison, Dennett's theory of the intentional stance challenges the existence of intrinsic intentionality, even in human beings. To him, all intentionality, including humans', is derived in the first place: "we [humans] are artifacts ... designed over the eons as survival machines for genes.... So our intentionality is derived from the intentionality of our 'selfish' genes" [6, pp.298]. Hence there is no fundamental difference between the intentionality of a computer system and that of a human being, for both are derived by those who interact with them.

Dennett's theory legitimizes system intentionality and helps to explain why many digital artifacts appear to be intentional to us. However, its limitation is that it does not completely explain why and how certain artifacts seem more intentional than others. As a further step toward looking at the analysis of code in socio-cultural context, in this paper we turn our attention to system authors' role in narrating system intentionality, particularly to the use of intentional vocabulary. Two pieces of existing work are of particular relevance here. Hayles [10] has noticed the significance of such narration in Alife, a domain that is very similar to intentional systems; Agre and McDermott have separately commented the role of intentional vocabulary in the practice of AI. After a brief account of related existing work, this section discusses how our approach extends them.

## 2.1 Lessons from Alife

Artificial Life (Alife), sometimes also referred to as "ALife," "alife," or "AL," bears many resemblances to AI. Above all, the goal for researchers from both areas is to instruct computer programs to display phenomena that are not commonly associated with machines — whether aliveness or intelligence. Hayles's existing work on Alife therefore is of particular use for us to unpack system intentionality.

In order to understand "[h]ow is it possible in the late twentieth century to believe, or at least claim to believe, that computer codes are alive — and not only alive, but natural," Hayles approaches her research question "by looking not only at the scientific content of the programs but also at the stories told about and through them" [10, pp.224]. Subsequently, Hayles examines these "narratives" at three levels. The first level includes "representations, authorial intention, anthropomorphic interpretation" of Alife computer programs. By observing how Alife researchers construct the narratives so that they are tightly interwoven into the operations of the program through terms such as "mother cell," "daughter cell," "ancestor," Hayles argues that "the program operates as much within the imagination as it does within the computer." Narratives at the second level, in comparison, are concerned with Alife as a legitimate research area within theoretical biology. In the pursuit of this goal, Alife programs need to be framed as *life-as-it-could-be*, containing the special case of *life-as-we-know-it* defined by the traditional biology. At the third level exist narratives of the relationship between Alife and the present and future of terrestrial evolution forms. Alife, rendered by such narratives, is not a simulation of the human, but rather becomes a model to understand the latter (pp.224-239).

The significance of Hayles's work is that it reveals that Alife is far from merely a technical practice. What she calls the "multilayered system of metaphors and material relays through which 'life,' 'nature,' and the 'human' are being redefined" is revealing to our own work of AI and system intentionality. It is also important, however, to clarify that we do not intend to repeat Hayles's work here, but rather to extend it by addressing the following aspects. First, we pay closer attention to the technical practice of AI. In addition

to interviews, talks, and presentations, upon which Hayles's analysis is primarily based, we intend to reveal the discursive nature of AI engineering as the root of system intentionality. Second, Hayles's analysis is ambivalent about the role of such narratives. It seems to imply that the discursive narrations exist independent of the actual, technical practice of Alife as a vigorous research area. As we will argue in sections 3 and 4, the technical practice of AI is intrinsically discursive.

## 2.2 The "Epidemic" of Intentional Vocabulary

Throughout the history of AI, new technological innovations have brought in a large intentional vocabulary to the field such as "reasoning," "planning," "learning," "strategizing," and "creating." These intentional terms are so pervasive in AI that it is impossible to talk about any algorithms or systems without using them. To the practitioners in the field, these terms have very specific meanings relating to specific methods, which are only "roughly correspondent" to their commonsense meanings. Are these words misleading and detrimental to AI practice? This section draws on the observations of AI practitioners, including Agre and McDermott, argue that the intentional terms are a constitutive component of AI practice.

Agre acknowledges the "dual character of AI terminology" is that these keywords are simultaneously formal and vernacular, enabling the practitioner to achieve "a sense of accomplishment" and to pin down precise structures and processes [2]. On the one hand, he denies that the strategic elasticity of these key terms as a conscious deception. On the other hand, he admits such use of intentional vocabulary is "self-defeating" because these terms inevitably link AI to a much larger discourse based on reflections of their vague meanings. The consequence is that AI practitioners "find it remarkably difficult to conceptualize alternatives to their existing repertoire of technical schemata" [2].

Drew McDermott [20] made a very similar, but more radical, observation on this issue two decades before Agre. He criticized the relationship between the formal and vernacular meanings of intentional vocabulary as "wishful mnemonics," and saw it as "a major source of simple-mindedness in AI programs." Identifying the use of these intentional mnemonics in a wide variety of AI systems ranging from the *General Problem Solver* (*GPS*) to language "translation" systems, McDermott warned AI practitioners that the epidemic of "contagious wishfulness" is misleading, most prominently to the practitioners themselves. Instead of naming their programs "UNDERSTAND" or "THINK," all disciplined programmers, he urges, should refer to their program as "G0034" and see if they can still convince themselves or anyone else that G0034 implements some part of understanding.

Appreciating the approaches of Agre and McDermott, our work differs in its strong focus on the active role of AI practitioners. This work disputes widely held notions of science requiring the independence of a system's operation from its authors' subjective explanation. Under such rhetoric, any disciplined practitioners are neural devices immune to their own "subjectivity." However, the dilemma here is that nature cannot speak directly. As Latour [17, pp.70-74] cogently argues, being the *spokesperson* for what is inscribed by her instruments is part of a scientist's mission. An "G0034"-styled program without the narration of its author is like an incomplete experiment, waiting for the scientist to be its "mouthpiece."

The examples of "UNDERSTANDING" and "G0034" are both extreme. In most cases, AI practitioners are simultaneously the executor and narrator for their systems. On the one hand, the formal meanings of many key intentional terms have been established and followed by AI practitioners in their systems. Simply naming a program "PLANNER" does not automatically legitimize it as a planner in the AI sense. In this sense, an AI practitioner (only) executes the conventions and methods allowed by their community of practice. On the other hand, the operations of the systems need to be narrated. Similar to the Alife researchers above, AI practitioners are engaged in the task of creating artifacts with properties that are not commonly associated with them before. The elasticity of the intentional vocabulary hence provides AI practitioners with an effective discursive device to close the gap between the operation of a system and the properties it is required to exhibit. We argue that without the glue of intentional vocabulary used in practitioners' narrations, the empire of AI would collapse. Therefore, the "wishfulness" is "contagious" not because it is "deceptive" but because it is *necessary* to the practice of AI.

# 3. THE AI HERMENEUTIC NETWORK

Informed by the works of Searle, Hayles, and Agre, we propose our new framework of the AI hermeneutic network (Figure 1). It emphasizes that system intentionality arises from a hermeneutic communication process, which incorporates two equally important components: the system author's discursive narrating and the user's hermeneutic reading in their respective contexts, negotiating with each other through actual system (e.g., source code and interface) and literature about the system (e.g., technical publication, media coverage, and authors' blogs) [27]. In this article, we focus on the author's narration, in particular their use of intentional vocabulary, and the social nature of the exchange of meanings between the author and the user.

The action of narration in the setting of a technical field, however, is not as straightforward as it may seem. We first need to differentiate an AI practitioner's narration of her system's intentionality from a kind of subterfuge story that obscures rather than explains system function. In the latter scenario, like a fairy tale that an adult constructs to cheer up a tearful child, the narrative is constructed from knowingly counter-factual materials in order to achieve a specific goal. The narration of system intentionality, on the other hand, is seamlessly integrated into the everyday practice of AI in the form of what Agre calls the "elasticity of meaning" that these AI keywords afford [2]. When a practitioner claims that her system is capable of "planning", what is at work is that the term's formal meaning temporarily takes over its vernacular signification. When a lay user, or sometimes an AI practitioner herself, encounters the discourses of the system, she may take on the vernacular meaning of "planning." This oscillation between formal and vernacular meanings is far beyond the binary boundary between "factual" and "counter-factual," a notion derived from a ro-

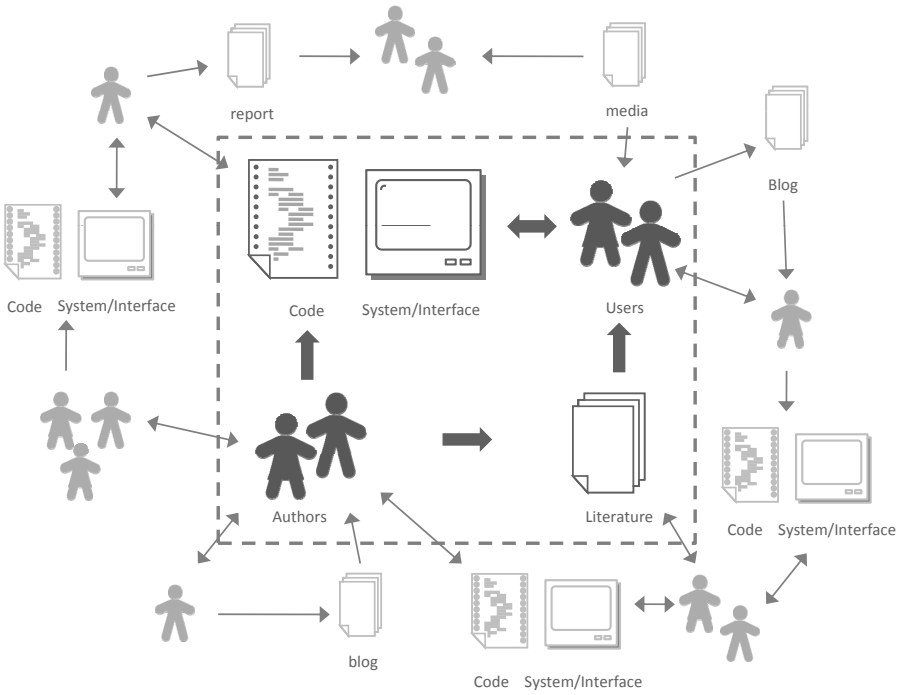



**Figure 1: The AI Hermeneutic Network**

manticized notion of science.

Our definition of system authors' narration is not limited to interviews, presentations, and other forms of inter-personal communications. As the case study in the next section will illustrate, such discursive narrations also manifest themselves through the technical construction of AI systems. Agre argues that "the purpose of AI is to build computer systems whose operation can be narrated using intentional vocabulary," particularly through AI keywords such as "planning" [1]. We extend this work by looking into the discursive machine through various other forms, including the choices of function, algorithm, system architecture, etc.

Briefly, users bring their own experiences and social and cultural backgrounds when they interact with systems in order to appropriate their meanings. In information studies, for instance, researchers [3] have conducted ethnographic studies of how users hermeneutically read quantitative data provided by information systems and how they contextualize these "cold and objective categories and numbers" with the real-life situations.

Finally, we acknowledge the impact of other social agents, referred to as actants in actor-network theory, in the meaning exchange process between the author and the user. Government funding agencies, media corporations, public relations managers for the institution where the system was built, an economic crisis, a technological breakthrough, etc., are all part of the network in which the hermeneutic communication of system intentionality takes place. More in-depth articulation of this framework can be found in Zhu's dissertation [25].

# 4. CASE STUDY: A CLOSE READING OF COPYCAT

In order to demonstrate the utilities of the AI hermeneutic network, this section presents a close reading of *Copycat*, a real AI system. The emphasis is its authors' narrative strategies in the construction of *Copycat*'s system intentionality, in particular the use of intentional vocabulary. The analysis of the program's source code and a substantial corpus of technical literature on Copycat demonstrates how its creators mobilize different discursive strategies to construct Copycat as an intentional system using both intentional and technical narrations. In particular, we emphasize the crucial role of the intentional vocabulary, which gives rise to system intentionality. Our reading reveals that the intentional vocabulary serves as a joint connecting the discursive needs and the technical requirements of the AI system.

The aim of the *Copycat* project is to model human "mental fluidity" in analogy making. Its authors claim that the program is capable of making "insightful" analogies in a small, restricted domain, namely alphabetic sequences. A typical example is **abc** → **abd**, **pqr** → **?**. In this case, *Copycat* may answer **pqs**, after it replaces the last letter "r" with its successor. However, these problems are not always as straightforward. One interesting feature of this domain is that many solutions may be valid for each problem, depending on how the subject (either human or computer) "interprets" the problem. For instance, when faced with a new problem **abc** → **abd**, **pqqrrr** → **?**, *Copycat* may provide answers of **pqqrrs**, **pqqsss**, or **pqqrrrr**. When the results are shown to humans, many find the last one the most "insightful" and "fluid" because the system mapps *alphabetic position* to *group size*.

## 4.1 The Corpus and Potential Limitations

The *Copycat* project was developed by Douglas R. Hofstadter, author of the 1980 Pulitzer-winning book *Gödel, Escher, Bach* [11], and his Ph.D. student Melanie Mitchell between 1984 and 1995. In addition to its source code, we include a substantial collection of related major technical papers and book chapters by the authors [12, 13, 14, 21]. These publications of over 200 pages constitute our primary corpus of analysis. We also incorporate additional material of Hofstadter's interviews with the mass media, non-technical articles, and personal websites, all of which provide us the social contexts of the project and the authors' ideological/philosophical positions on issues related to intentional systems.

Admittedly, several potential limitations exist. First, since the corpus does not exhaustively include all publications on *Copycat*, certain discursive strategies could be used differently or completely left out of our analysis. Second, other authors' narration may differ from those discussed here. Although these issues may potentially undermine the generalizability of our analysis, our aim is not to propose a generic pattern that fits all AI practitioners. This is where detailed context and ideological analyses in [25] become very useful. This paper highlights, through the example of *Copycat*, system authors' narrations as a crucial source of system intentionality.

## 4.2 The Two Languages of Copycat

In our reading of the primary corpus, we identify two paralleling languages used simultaneously in the technical literature of *Copycat*. These two languages, one intentional and the other technical, are intertwined with each other and complement each other in Hofstadter and Mitchell's narrations of system intentionality. In order to draw contrast to these two different discursive strategies, we first artificially separate them into two different narratives of the system's operation. By doing so, however, we do not suggest the existence of an "objective" technical language and another discursive one, independent of each other. As argued above, the technical practice of AI is intrinsically discursive. This artificial separation thus is our strategy to draw readers' attention to the coexistence of two different "semiotic systems" [19].

### 4.2.1 A Stochastic Local Search Program

We start by looking at the technical narration of *Copycat*. *Copycat* is a stochastic local search program. It receives three character strings (String 1, String 2, and String 3) as input and generates a single output character string. During the process, *Copycat* performs a stochastic local search in a particular search space, optimizing a particular heuristic function. The search space is the space of all possible structures that relate the three input strings together. Each one of these structures is a graph built from a base set of primitive constructs predefined by *Copycat*'s authors (such as "b is the successor of a"). A particular structure captures the relations among the three input strings and determines the compatibility of the different primitive constructs appearing in a structure. *Copycat* maximizes a heuristic function, that is, the extent to which the proposed structure captures all the regularities and relations among the three strings. The system may randomly terminate its search at any point in time. The probability of termination is higher if the system has found a structure with a high value based on the heuristic function. Once the search stops, the system generates an output string according to the transformation operations specified by the current structure. This means that the same operations that transform String 1 to String 2 will be applied to String 3 in order to derive the output string. For instance, when *Copycat* receives the following input: **abc**, **abd**, and **pqr**, its answer will be **pqs** with a high probability.

### 4.2.2 A Fluid Analogy Maker

Intermixed with the above technical language is the intentional narration of *Copycat* as a fluid analogy maker, focusing on the program's psychological plausibility and applicability to the related human mental process. The system models the "mental fluidity" in the human analogy-making process and constructs "insightful" analogies. In particular, *Copycat* implements the "slippage" of concepts from one into one another. In the previous example, we say the concept of "alphabetic order" slips into "group size." The system consists of three modules: a "slipnet," locus of all the concepts that *Copycat* has access to; a "workspace," where the system constructs representations of its current problem and computes the final results; and finally a "coderack," which contains a collection of *codelets* is waiting to be executed with its respective probability. Codelets are small pieces of code that perform various tasks such as creating or destroying a new perceptual structure, evaluating how promising a particular structure is, or creating more codelets. They can be seen as the enzymes in biological cells, where each enzyme does only one very small task, but the combination of thousands of them manages to fulfill complex tasks.

## 4.3 The Use of Intentional Vocabulary

The coexistence of the two languages in the technical literature is far from a coincidence. It speaks to the issues that *Copycat*'s authors hope to address. Compared with other computational analogy systems (e.g., *SME* [9] and *ACME* [15]), Hofstadter and Mitchell claim that their system exceeds the others in two main aspects. First, it models both the perception and mapping stages of analogy, whereas other existing systems only tackle the second stage. Second, its biologically inspired search scheme is more "psychologically plausible" in comparison with the traditional exhaustive search methods used by other models of analogy. The system's resemblance to human cognition and the technical requirement of the system's operation naturally lends themselves to each of the two languages. But how do the technical and intentional languages connect to and build upon one another?

We argue that the intentional vocabulary serves as the joint between the two languages and gives rise to system intentionality. Before illustrating our point with an example, we first identify the authors' three main strategies of using intentional vocabulary. First, intentional verbs are heavily used to narrate the system's operation. Such words as "know," "resist," "understand" appear throughout the primary corpus. More examples include (emphasis added):

> Just as the program *knows* the immediate neighbors of every letter in the alphabet, it also *knows*

> the successors and predecessors of small integers.

> Copycat tends to *resist* bringing numbers into the picture, unless there seems to be some compelling reason to do so.

> Musing codelets allow several different and rival pathways to be *sniffed* or *checked out*.

Second, certain data structures and functions are titled with human cognitive faculties and human mental states. If the previous strategy is concerned with the intentionality of the system, the narrations at this level are intended to draw close connection between *Copycat*'s operation to human cognitive process and lay the groundwork for the next strategy. For instance, the search space of *Copycat* is composed of structures called "point of views" (or simply "views"), which specify the ways different "concepts" connecting the three input strings (e.g., **abc**, **abd**, and **pqr**). Similarly, *Copycat* has "long-term memory," "drive," "desire," and "personality." Below are more examples (emphasis added):

> It [(the Slipnet)] can be thought of, roughly, as *Copycat*'s *long-term memory*.

> [*Copycat*] must reconcile a large number of mutually incompatible local *desires* (the technical term for this is '*frustration*').

> ... and those data provided some of the most important insights into the program's "*personality*."

Last but not least, the system is often narrated in comparison to human and other forms of life (e.g., cells, and ants). Although the content may vary depending on the context, these arguments typically take the following form: A (creative) human faced with situation X will react with action Y, and *Copycat* also performs action Y in this situation X. The purpose of drawing such a comparison is to imply that Copycat is similarly creative, intelligent and intentional. An example is:

> In particular, people are clearly quicker to recognize two neighboring objects as identical than as being related in some abstract way. Thus the architecture has an intrinsic speed-bias in favor of sameness bonds: it tends to spot them and to construct them more quickly than it spots and constructs bonds representing other kinds of relationships.

## 4.4 The Vocabulary of "Happiness"

This section provides an example of how the use of intentional vocabulary connects to both the technical and intentional narration of the system. Most AI practitioners will be quick to agree that "happiness" is an explicitly intentional term. Unlike terms such as "planning" or "learning," "happiness" does not have a conventionally agreed upon formal meaning that the AI community follows. In fact, its highly subjective and emotional undertone is almost an antithesis of any formal definition based on machine operation. This section provides an example of how the authors of *Copycat* use intentional vocabulary such as "happiness" to connect the intentional narration needed for their research goal and the technical requirement for the systems.

In *Copycat*, the level of "happiness" of each object (e.g., a letter, a number, or a group of letters from the three input strings) in the system's "work space" is an important factor. It regulates how much attention the system pays to a specific object; the unhappier an object is, the more resources will be given to it. As its authors describe,

> Unhappiness is a measure of how integrated the object is with other objects. An unhappy object is one that has few or no connections to the rest of the object in the Workspace, and that thus seems to cry out for more attention.

The choice of an emotional term with easily understandable meaning is far from an accident. It reinforces the overarching research goals set by *Copycat*'s creators, that is, to model the analogy-making process with psychological plausibility. As the authors position the system in one of the opening paragraphs, "Copycat is a computer program designed to be able to discover insightful analogies, and to do so in a psychologically realistic way." One effective strategy to accomplish this goal is to connect the system's operation to common wisdoms. The authors wrote: "the architecture follows the old motto 'The squeaky wheel gets the oil', even if only probabilistically so." Following this motto, it seems "natural" and "human-like" that the "unhappy" concept is entitled to more attention.

However, the "unhappiness" measure also serves a technical purpose that is never explicitly mentioned in the corpus. Technically speaking, *Copycat*'s goal is to optimize the overall connection between different objects in the working space. The strength of each structure is computed as an aggregation of the strengths of the individual elements (e.g., bonds) in the structure. An "unhappy" element corresponds to an element with weak structures, whereas a "happy" one has an already formed strong structure that connects it to other elements. In probabilistic terms, working on the weakest point in a structure yields the most chances for improvement; for modifying a strong structure is likely to make it weaker (since the structure is strong), and modifying a weak structure is likely to make it stronger (since it is already weak). Hence, focusing on the "unhappy" objects maximizes the probability of strengthening the current structure.

Figure 2 provides the source code for one particular kind of "happiness." The implementation shows a rather simple function with little connection to the vernacular meaning of the term. Here, the so-called "intra-string-unhappiness" is updated with a number between the range of 0 and 100, depending on the properties of the connection this element has. As stated earlier, the point of the example is not the

```
(record-case (rest msg)
  ...
  (update-intra-string-unhappiness ()
    (set! intra-string-unhappiness
      (cond
        ((tell self 'spans-whole-string?) 0)
        ((exists? enclosing-group)
          (100- (tell enclosing-group 'get-strength)))
        (else
          (let ((bonds (tell self 'get-incident-bonds)))
            (cond
              ((null? bonds) 100)
              ((or (tell self 'leftmost-in-string?)
                   (tell self 'rightmost-in-string?))
                (100- (round (* 1/3 (tell (1st bonds)
                                          'get-strength)))))
              (else
                (100- (round (* 1/6
                                (sum (tell-all bonds
                                               'get-strength))))))))))
    'done)
  ...
)
```

**Figure 2: Source Code for One Kind of "Happiness"**

disjunction between the intentional narrative and the computational operation. Rather, it illustrates the intentional vocabulary's pivotal function of connecting the two. Without the intentional narration, the technical machine operation lacks the system intentionality necessary for intentional and AI systems. Without the machine operation, on the other hand, intentional narrations are vague and hard to believe. The intentional terms such as "happiness" provide a joint so that the discursive and computational operations can cling to each other.

# 5. CONCLUSION

In summary, this paper has proposed a humanistic and interpretive framework to analyzing intentional systems through our new construct of the AI hermeneutic network. Different from seeing system intentionality as an intrinsic (technical) property of software, we highlight an actor-network of which software is just one component. The central analysis here includes both source code and technical literature, as a location for the meaning making process between system authors and users in its social context. The technical literature surrounding software systems so far has been a relatively unexplored area in software studies.

By applying this framework to a real AI system, we have identified various discursive strategies that the system authors used to narrate the system intentionality of *Copycat*. More importantly, such narrations are part of the technical practice of AI. The authors' various uses of intentional vocabulary, as we have seen, connect the discursive and technical requirements of the system. In this regard, the practice of AI is fundamentally technical and discursive at the same instant. This often-neglected discursive aspect of how we understand system functionality stresses the importance of critical theories and humanistic values to understand the construction of AI systems.

As part of our future work, we plan to further apply the AI hermeneutic network to other AI systems. We also intend to explore expanding the framework to the analysis of instrumental software, and software in general. Certainly, many of the issues pertinent to AI can also be applied to the broader domain of software and the burgeoning area of Software Studies explored by researchers such as Lev Manovich, Noah Wardrip-Fruin, Mathew Fuller, Jeremy Douglass, Mark Marino, and others point to a recognition of the need for software studies methods. Part of our contribution is to critically analyze the practice of AI from the vantage point of an insider-outsider. Just as ethnographer who, when living in a different culture, must (ideally) become a member of the group being studied, our approach is based on our experiences as practitioners in the community of AI. At the same time, as the ethnographer is essentially an "outsider" and inevitably makes sense of this culture through an external lens necessitated by framing her results as research, our critique of, and approach to, AI practice and intentional systems are informed by the external lens of theories of the humanities and social sciences.

# 6. REFERENCES


[1] P. E. Agre. *Computation and Human Experience*. Cambridge University Press, Cambridge, U.K., 1997.

[2] P. E. Agre. Toward a critical technical practice: Lessons learned in trying to reform ai. In G. C. Bowker, S. L. Star, W. Turner, L. Gasser, and G. Bowker, editors, *Social Science, Technical Systems, and Cooperative Work: Beyond the Great Divide*, pages 131–158. Lawrence Erlbaum Associates, 1997.

[3] R. J. Boland. Information systems use as a hermeneutic process. In H.-E. Nissen, H. K. Klein, and R. Hirschheim, editors, *Information Systems Research: Contemporary Approaches Emergent Traditions*, pages 439–458. North-Holland, New York, 1991.

[4] F. Brentano. *Psychology from an Empirical Standpoint*. Routledge & Kegan Paul, London, 1874.

[5] D. C. Dennett. True believers: the intentional strategy and why it works. In j. Haugeland, editor, *Mind Design II: Philosophy Psychology Artificial Intelligence*, pages 57–79. MIT Press, 1981.

[6] D. C. Dennett. *The Intentional Stance*. MIT Press, Cambridge, 1987.

[7] D. C. Dennett. *Kinds of Minds*. Basic Books, New York, 1996.

[8] D. C. Dennett and J. Haugeland. Intentionality. In R. L. Gregory, editor, *The Oxford Companion to the Mind*. Oxford University Press, Oxford, 1987.

[9] B. Falkenhainer, K. D. Forbus, and D. Gentner. The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, 41:1–63, 1989.

[10] N. K. Hayles. *How We Became Posthuman: Virtual Bodies in Cybernetics, Literature, and Informatics*. University of Chicago Press, Chicago, 1999.

[11] D. R. Hofstadter. *Gödel, Escher, Bach : an eternal golden braid*. Basic Books, New York, 1979.

[12] D. R. Hofstadter. The copycat project: An experiment in nondeterminism and creative analogies. Technical Report AI Memo No. 755, AI Laboratory, MIT, 1984.

[13] D. R. Hofstadter and M. Mitchell. The copycat project: A model of mental fluidity and analogy-making. In J. A. Barnden and K. J. Holyoak, editors, *Advances in connectionist and neural computation theory: Analogical connections*, volume 2, pages 31 – 112. Ablex, Norwood, NJ, 1994.

[14] D. R. Hofstadter and M. Mitchell. The copycat project: A model of mental fluidity and analogy-making. In D. R. Hofstadter, editor, *Fluid concepts & creative analogies*, pages 205–268. Basic Books, New York, 1995.

[15] K. J. Holyoak and P. Thagard. Analogical mapping by constraint satisfaction. *Cognitive Science*, 13:295–355, 1989.

[16] E. Husserl. *Logical Investigations*. Routledge, 1970.

[17] B. Latour. *Science in Action: How to Follow Scientists and Engineers Through Society*. Harvard University Press, Cambridge, 1988.

[18] M. C. Marino. Critical code studies. *Electronic Book Review*, http://www.electronicbookreview.com/thread/electropoetics/codology, 2006.

[19] M. Mateas. *Interactive Drama, Art, and Artificial Intelligence*. PhD thesis, CMU, 2002.

[20] D. Mcdermott. Artificial intelligence meets natural stupidity. *SIGART Bull*, 57:4 – 9, 1976.

[21] M. Mitchell and D. R. Hofstadter. Perspectives on copycat: Comparisons with recent work. In D. R. Hofstadter, editor, *Fluid Concepts & Creative Analogies*, pages 275–299. Basic Books, New York, 1995.

[22] J. Searle. *Intentionality: An Essay in the Philosophy of Mind*. Cambridge University Press, Cambridge, 1983.

[23] N. Seel. *Agent Theories and Architectures*. PhD thesis, Surrey University, 1989.

[24] T. Winograd and F. Flores. *Understanding Computers and Cognition*. Ablex Publishing Corporation, Norwood, NJ, 1986.

[25] J. Zhu. *Intentional Systems and the Artificial Intelligence Hermeneutic Network: Agency and Intentionality in Expressive Computaional Systems*. PhD thesis, Georgia Institute of Technology, 2009.

[26] J. Zhu and D. F. Harrell. Daydreaming with intention: Scalable blending-based imagining and agency in generative interactive narrative. In *AAAI 2008 Spring Symposium on Creative Intelligent Systems*, pages 156–162, Stanford, CA, 2008. AAAI Press.

[27] J. Zhu and D. F. Harrell. The artificial intelligence (ai) hermeneutic network: A new approach to analysis and design of intentional systems. In *Proceedings of the 2009 Digital Humanities Conference*, pages 301–304, 2009.