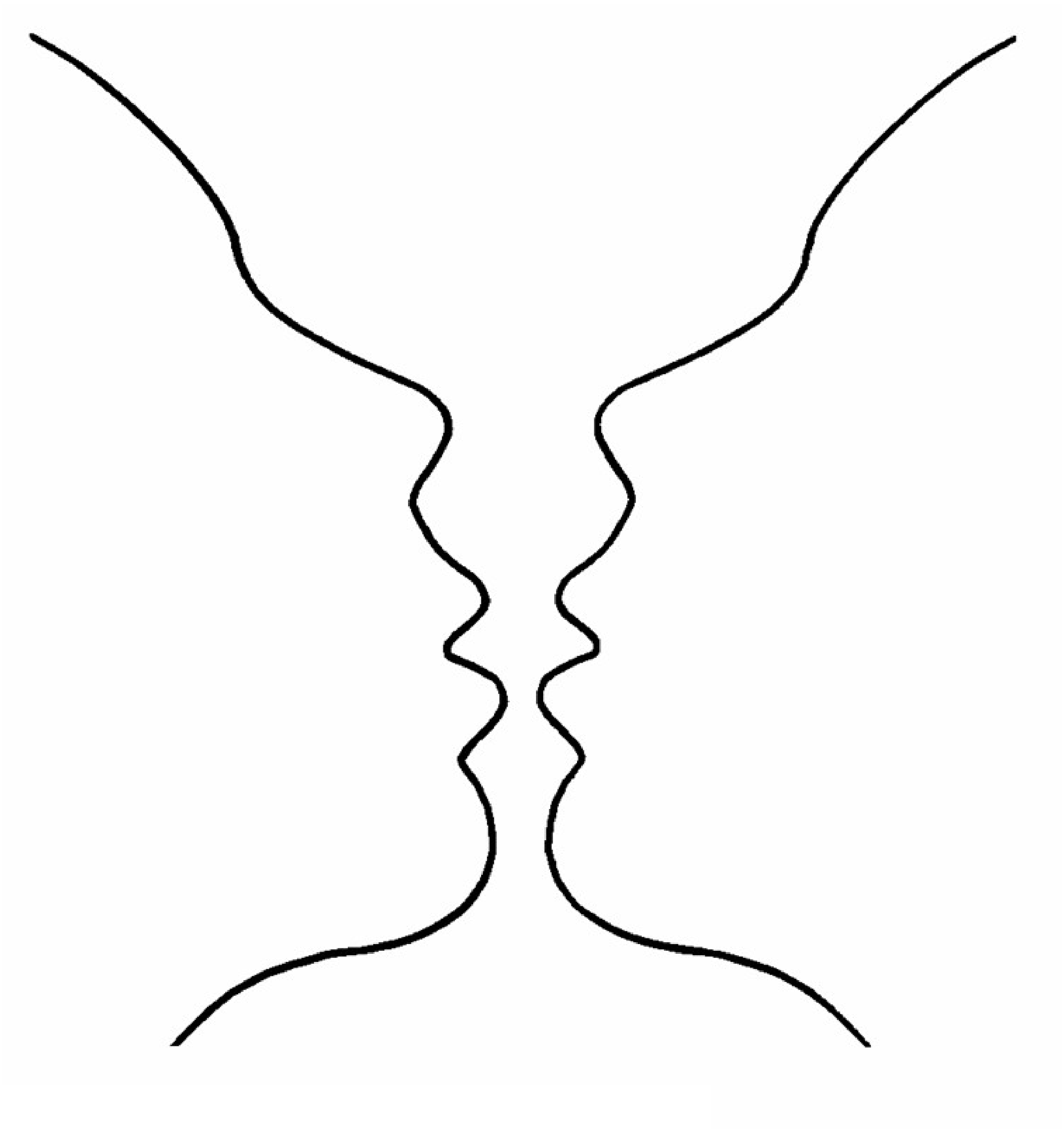
Emerging Symbols



S. Leijnen - 0140732
December 2007
INF/SCR-07-33

Table of Contents

Abstract

At the heart of many challenges facing modern-day Artificial Intelligence lies the Symbol Grounding problem, a theoretical impediment for computers to understand the meaning of symbols. It is argued that dealing with this problem begs for a multi-disciplinary approach, drawing from a diversity of fields such as philosophy, sign theory, computer science and biology. The various denotations of symbols are meticulously analyzed, uncovering a hierarchy in the human interpretation process. It is explained how symbolic interpretation emerges from sub-symbolic components and how this transition may be modeled using a neural network simulation. A series of experiments is conducted in order to validate the capabilities of this new model.

Before you study Zen, mountains are mountains and rivers are rivers; while you are studying Zen, mountains are no longer mountains and rivers are no longer rivers; but once you have had enlightenment, mountains are once again mountains and rivers again rivers.

– Zen koan

Introduction

Throughout history, various technological innovations have been treated as metaphors for the human brain. The mechanics of a steam engine, the accuracy of a watch, and more recently the interconnected computer grid that forms the Internet have all been compared to our own thought processes. But, more than any other device, the computer has been and remains to be the most convincing example of a mechanical brain. The similarities are striking: vast amounts of small processing units guiding electronic signals, translating input to output or, in the case of robots, mechanical behavior. It manages to solve algebraic equations, play a game of chess or fly an airplane. Although there are currently still many areas of expertise where humans outperform computers, it seems that given the right program and a robotic body, a computer can do almost anything we can do.

In fact, the computer and the brain are so alike, that people have started to ask questions about the nature of computational minds. Are these machines doing the same thing we are doing? Do they really understand their actions? Are they conscious beings, like us? Although these kinds of questions often originate out of philosophical curiosity, finding the answers is of no less practical importance to Artificial Intelligence research. Several technical difficulties, many of which have surfaced over the past decades, can be directly attributed to a lack of understanding of these philosophical issues.

Specifically, the question whether computer programs are capable of meaningful symbol interpretation has received much attention, and it is often suggested that having such a capability is a necessity for a computer to take part in a natural language conversation. For example, Terry Winograd notes that

> *"Programs can manipulate linguistic symbols with great facility, as in word-processing software, but attempts to have computers deal with meaning are vexed by ambiguity in human languages"*
>
> (Winograd, 1984, p. 102)

Considering the important role of language in human intelligence, one would naturally expect *meaning* to be a significant part of any artificial intelligence program. However, this is easier said than done: when one actually undertakes the effort to build a computer program capable of meaningful interpretation, several serious obstacles are encountered. The first chapter of this thesis introduces a few of these problems and discusses two notable objections in particular: the Chinese Room Argument (Searle, 1980) and the Symbol Grounding Problem (Harnad, 1990). These two ideas will recur throughout the thesis as they are canonical exponents of the underlying central theme: the problem of computers dealing with meaning.

Although its name might suggest the possibility of a solution, the Symbol Grounding Problem is in fact a fundamental constraint on the interpretation skills of computers - it cannot be solved *an sich* (Vogt, 2002). That is why we will approach the problem from a different angle, and investigate whether meaningful interpretation is possible in other systems than the traditional rule-based symbol systems. Hence, our main research question will be as follows:

> *"How are symbols to be represented, and how do we show the advantages of symbolic representation?"*

A complex question like this begs for a thorough investigation. We will therefore approach it from several disciplines and search for the general foundations of *symbolic interpretation*. To this end, we will acquire a clear understanding of what symbols are. In chapter two, we learn that the word *symbol* is a rather ambiguous term; by clarifying its meaning we find a good starting point for further investigation. While the notion *symbol* refers to a simple correspondence between two tokens in some contexts, in other settings it denotes a more obscure relation that is subject to the knowledge of the interpreter. It turns out that these two meanings collate with the notions *icon* and *symbol*, respectively, within the philosophical framework of C. S. Peirce (Deacon, 1997). We examine his theory of signs and interpretation further in order to gain an understanding of the nature of symbols.

It follows from Peirce's sign theory or *semiotics* that icons and symbols are not merely two distinct types of signs; the third chapter investigates how they are linked. Using several examples we demonstrate how three kinds of signs are related within a hierarchical structure. Symbols are constructed from *indices*, the third type of Peircean sign. Icons are a necessary part of indices. At this point, we have found the blueprints for a symbolic interpreter, allowing us to leave the philosophical domain and return to the original question of representing symbols.

In chapter four, we look for an application of this theory in order to answer the first part of the research question. It is argued that *neural networks* are a fitting model for realizing symbolic representations. Following an overview of their structure and mechanisms in chapter four, we rephrase the hierarchical ordering of signs from the previous chapter to match the neural network architecture. Each of the signs is translated to a specific neural network, resulting in concrete models of an iconic, an indexical and a symbolic interpreting system.

In order to answer the second part of the research question, we need to compare the learning capabilities of the indexical and symbolic neural network models. Aiming to validate our theory, we turn to a chimpanzee language training study in chapter five. It shows the difference between indexical and symbolic learning through analysis of the chimps' token manipulation: two different learning strategies emerge, each with its own learning curve (Savage-Rumbaugh, 1978). By translating the elements of this study to a setting that is fit for neural networks and testing the neural network models in a series of experiments, we too can discern a difference in learning and demonstrate the advantages of symbolic interpretation.

In the final chapter of this thesis, the results are summarized and the implications of this research are discussed.

Chapter 1: The Symbol Grounding Problem

The Chinese Room Argument

In his famous article *Minds, Brains and Programs* (Searle, 1980) the philosopher John Searle invites us to take place inside a computer and imagine what it is like to be one. We find ourselves in a room entirely sealed off from the outside world, except for a small window not unlike the iron framed vent found in many Chinese restaurants through which food is handed to the waiters. In this case, the window is used to deliver data in the form of a written note containing a question. The question is posed in a language we are unfamiliar with, Chinese for instance. Since we have no knowledge of that language, we will not be able to understand the question - let alone produce a correct answer in Chinese.

Fortunately, on a table inside the room we find a book, with on each page a set of rules like a sort of dictionary. A normal, Chinese-English dictionary would allow us to translate the question, formulate an answer in English and translate it back to Chinese. However, this particular dictionary contains only rules that point from Chinese characters to other Chinese characters. The rules have the interesting property that repeatedly applying them to the characters on the question note will always lead to a correct answer, in Chinese. Answering questions has been reduced to an almost trivial execution: all we have to do is apply the rules and hand back the results through the vent.

The thought experiment above brings Searle to make his point: do we usually answer questions this way, or is there a difference in our understanding of the question? It appears that using such a rulebook is quite an odd way to answer a question. We generally know what questions are about in order to answer them. English words have meaning to us, and by virtue of their reference to other concepts we manage to formulate a reply. Chinese words lack any meaning; we can only discern them by noticing the different lines and dots that together form a character. Given an external source of reference containing rules for each of these characters, we *can* use our ability to discern them to apply the right rules, without having the slightest idea what the question is about. But is this discriminatory ability the same as a genuine understanding of symbols? Maybe the person on the other side of the vent, who distributes questions and receives back our notes, thinks we do know Chinese. Surely, it must seem likely from that point of view that the characters have reference for us. However, likely as it may be, it is not necessarily the case.

The same argument can be applied to computers. A program consists of a fixed set of rules, executed by a computer to maintain a state or provide a certain output. For example, we can design a simulation program that predicts the probability of traffic jams based on a few parameters like the average speed or the number of traffic lanes. There is no need for the computer to get into the details of reality here: the rules and numbers that refer to cars, roads or weather conditions to humans, are merely mathematical tokens for the computer. It can simply apply a number of abstract mathematical rules that are defined within the program, ending with a probability estimate. Instead of an iron vent we update the parameters with several sensors placed near roads and use the program's output on road displays, and the analogy with the Chinese Room is complete. We have created a rule-based system that appears to understand and solve a problem, while in reality it only applies to a number of simple rules which were designed by someone who *did* have reference to the problem domain.

The possibility of a pseudo-intelligent system, which only simulates understanding, raises serious doubts about the validity of the claim that computational intelligence is similar to ours. From an outside perspective, how can we ever be sure of this claim without ruling out this possibility? For such a claim to be verified or rejected, a theory about the nature of computation is required that either explains the different modes of understanding Searle is referring to, or shows that all kinds of understanding are actually the same. Even if we choose to ignore the philosophical objections to the claim and just focus on practical applications, troublesome obstacles persist.

At a first glance, a lack of understanding in computers may appear to be of minor importance, since a well executed computation does not *require* any knowledge of what the data stands for. If a program would be equipped with enough procedures to appropriately handle any situation it encounters, we would probably not be able to discern it from an understanding system. So from a practical perspective, we might deduce that any differences in the level of understanding are trivial, provided that the output is correct. However, the assumption that a program *can* be designed to handle every possible situation simply doesn't hold for all domains. Given a setting with a constrained set of events and actions, the minimal number of rules required to handle every situation imaginable may still be relatively small. For example, the game *tic-tac-toe* has a very limited number of states, allowing for a full description of the optimal action in every possible state. Nevertheless, this solution only goes as far as the complexity of the task at hand allows. For games like chess or backgammon, where the number of possible states exceeds the computer's memory capacity, a different approach is clearly required.

Generally, as the domain within which the program acts grows larger, the complexity of the procedures will often follow. Programs that operate in unbounded domains, such as robotics or natural language, are particularly prone to encounter unexpected situations for which no rules exist. When communicating with a chat bot, one is likely to receive curious responses from time to time. One might say that the chat bot has apparently *misunderstood* these questions. However, this is a case of mistaken anthropomorphism: naturally, all questions are misunderstood by the program or, rather, not understood at all. A chat bot lacks any connection to the subjects that are discussed in the conversation, so it can't be said to either understand or misinterpret any questions. If anyone is to blame for the program's incorrect response to a question, it is the programmer who, in his attempt to simulate a conversing human, has failed to anticipate this particular question.

The occasions in which comparable problematic situations arise are plentiful, and so are the names that have been coined for these kinds of problems. The *Frame Problem* (McCarthy & Hayes, 1969), originating out of situation calculus, describes the situation where every single part of a robot's domain knowledge needs to be represented explicitly, even when things do *not* change as a result of an action. What we would consider to be trivial information is still required as a logical expression in the program, for example the rule that "a red box is still red when you put it on top of another box". It is closely related to the *Common Sense Knowledge Problem* (Dreyfus, 1981), the problem of representing the kind of implicit knowledge humans possess. Another canonical problem, which focuses particularly on the domain of symbolic knowledge, is called the *Symbol Grounding Problem* (Harnad, 1990). It deals with the nature of meaning and whether computers can understand symbols.

The manifestations of these and other, related problems express fundamental difficulties in the undertaking of modeling intelligence with computers. A related issue is raised by the Chinese Room Argument. In order for us to obtain a better understanding of the root of these problems we will examine the Symbol Grounding Problem more closely, as it is a recent formulation that expresses these difficulties in a particularly clear way.

The Symbolic Merry-go-round

When Alan Turing, one of the founding fathers of computer science, was looking for a way to test a program's intelligence, he devised the *Turing Test* (Turing, 1950). It involved a human subject having a conversation with either another human being or a computer program. Right after this dialogue, the subject would have to guess whether its conversation partner was a person or a computer. To avoid any predispositions derived from external features, such as the computer's appearance or word pronunciation, all conversations would appear on a monitor and the subject would have to type its responses using a computer terminal. This way, the anonymity of the conversation partner would be guaranteed and the subject's judgment would be based exclusively on the dialogue's semantic content. If a majority of the subjects repeatedly failed to make correct guesses, Turing concluded, then we would be forced to accept the computer program as an intelligent entity, as its conversation capabilities were undistinguishable from those of humans.

It is not a coincidence that both Turing and Searle chose natural language as the crucial testing ground for a comparison between humans and computers. The capability of any system to exert this typically human way of communication forms an ideal criterion for determining its level of intelligence: not so much because recognizing, writing and pronouncing symbols form such insurmountable barriers, but because a conversation can basically be about anything. The domain of language is unbounded, making the rule-based dictionary solution an insufficiently equipped approach for passing the test. To deal with all possible topics, a dictionary would have to include every single answer imaginable, making it practically infeasible. A more sophisticated solution would have to be found.

Turing and Searle both identify language as the decisive benchmark test, but whereas Turing is satisfied with empirical evidence as a proof for intelligence, Searle uses a more strict definition: only if a system understands language the same way we do – a functional criterion – will the claim for its intelligence be verified. When applied to the Chinese Room, both of these requirements fail to make any unambiguous claims, though. Searle strives to define the difference by pointing to the term *understanding*. However, without an operational definition of this term his argument remains philosophical, rather than scientifically verifiable. Switching to the external perspective of the Turing Test, the difference in understanding between using a dictionary and using domain reference is undistinguishable and therefore not relevant according to Turing. But it is questionable whether such a dictionary could really exist, given the problems with the domain of natural language that were discussed previously. What other approach would solve this problem?

Perhaps the most straightforward solution would be to put another dictionary in the room, containing translations from Chinese to English symbols and vice versa. The person inside is now capable of translating the characters into a familiar language, understand the question and answer it correctly without ever consulting the first dictionary. From the outside perspective of the Turing Test there would be no difference perceivable, while the approach itself is now viable: the previous dictionary containing all possible rewriting rules – a practical impossibility - is now replaced by a normal language-to-language dictionary. On top of that, Searle's critique would be avoided altogether, since the person uses his own, understanding intellect to answer the question.

Unfortunately, as Stevan Harnad argues in his article *The Symbol Grounding Problem* (Harnad, 1990), the problem is not that easily solved. A translation of the symbols of one language to another would certainly enable humans to hold an intelligent conversation. However, recall that the Chinese Room Argument compares computers with humans. The analogy becomes invalid once we invoke a capacity that is available for the one and not for the other. Humans already have a language in which their knowledge about the world can be expressed, but for a computer such a *grounded* language is not available. A computer needs to find the meaning of its own symbols from scratch, like children learning their first language. Translating unknown symbols into another meaningless language doesn't give them meaning; it only leads to infinite regress. As Harnad puts it:

> *"Suppose you had to learn Chinese as a first language and the only source of information you had was a Chinese/Chinese dictionary! This is more like the actual task faced by a purely symbolic model of the mind: How can you ever get off the symbol/symbol merry-go-round? How is symbol meaning to be grounded in something other than just more meaningless symbols? This is the symbol grounding problem."*
>
> (Harnad, 1990, p. 338)

So, simply grounding symbols in another language evades the real question at hand. The solution relies on knowledge not available in any computer. The only way to get off the symbolic merry-go-round Harnad refers to would be to make the semantic interpretation intrinsic to the system. Not by creating a program that manipulates symbols that have meaning to *us*, but by making sure the program *itself* knows what the symbols refer to. That way, it will be able to understand the meaning of its expressions and, following the claims of both Turing and Searle, it will demonstrate a form of intelligence that is comparable to ours.

Chapter 2: What are Symbols?

The Meanings of *Symbol*

In the previous chapter we have used the Chinese Room Argument to introduce the Symbol Grounding Problem, a central question in Artificial Intelligence research. The discussion that resulted from the argument's publication forms an excellent starting point for an analysis of the problem, as proponents and opponents have confronted each other with interesting reasonings to prove or disprove the validity of the thought experiment. Generally, opponents of the argument criticize Searle's claim that there are two different modes of understanding. It may intuitively feel like there is a genuine difference between using a rulebook for Chinese questions and answering an English question without one; but, according to many of the argument's critics, this is merely an illusion. They claim that the two modes of answering are actually one and the same and we are fooled by the subjective experience of understanding. This position assumes our brains can be modeled like the rules of a dictionary, implying a rather mechanistic view of the human brain. Some take it one step further and claim that humans have a kind of dictionary in their heads, with rules matching the symbols on the sheets. If so, then it is irrelevant whether these symbols are English letters or Chinese characters, since all that matters now is that they match with the brain's rule-based system.

Contrarily, Searle and his supporters argue that the brain does not work like a dictionary. A person who understands Chinese does not match symbols on a piece of paper to similar symbols in the head, after which he applies the corresponding rule. They claim that, apart from their physical appearance, the lines and dots that form Chinese characters carry meaning about something that may be entirely unrelated to paper or ink. These tokens are a set of carefully constructed lines and dots, drawn to convey a specific message. Of course, this view alone does not yet rule out the mechanistic position, but it does hint towards a different perspective on the process of understanding. Unlike a dictionary, where all meaning is encompassed in the rules, the brain actively connects symbols to what they stand for, giving them a form of reference beyond their merely being connected to other symbols. For someone who is familiar with a language, symbols stand for something. Because of this, he is capable of understanding a sentence.

Properly indicating the crucial point of difference between these two standpoints would require an accurate model of the process of understanding. However, lacking such a theory at this point, we can begin by analyzing the arguments put forward in the debate to expose the differences. Not surprisingly, the main point of discussion concerns the way symbols are processed. Symbols are either manipulated according to a rule matching their appearance, or they are connected to a concept they stand for and manipulated according to the logic of their referent. That the former is true when an Englishman is confronted with a Chinese dictionary is acknowledged by both parties; they only disagree whether, in the latter case, the person understands the symbols. Thus, the main distinction between the opposing parties is about the definition of symbols themselves. Are they simply perceived, recognized and manipulated accordingly? Or is there a more complex process involved, in which their referents play an important role? It appears that, in order to answer the question of how symbols can have meaning, these two meanings of the word *symbol* themselves need to be explained.

The two positions can be placed in a wider perspective: in his article *Universal Grammar and semiotic constraints* (Deacon, 2003), the American anthropologist Terrence Deacon argues that the definition of symbols in fields such as computer science, mathematics, cognitive science and recent philosophy deviates from the way the term is used in the humanities and social sciences. He uses the following characterizations:

> Computation: *A symbol is one of a conventional set of tokens manipulated with respect to certain of its physical characteristics by a set of substitution, elimination, and combination rules, and which is arbitrarily correlated with some referent.*
>
> Humanities*: A symbol is one of a conventional set of tokens that marks a node in a complex web of interdependent referential relationships and specific reference is not obviously discernible from its token features. Its reference is often obscure, abstract, multifaceted, and cryptic, and tends to require considerable experience or training to interpret.*
>
> (Deacon, 2003, p. 116)

These definitions call for further examination of the respective fields, to understand how and why they are rooted in them. Consider once again the principal target of Searle's criticism: the traditional symbol system approach to AI, expounded by Allen Newell and Herbert Simon (Newell & Simon, 1976). According to the computational definition of symbols, a program manipulates electronic bit tokens, governed by a fixed set of formal rules. The functional relations between these symbols are already explicitly defined in the rules, leaving no room for any additional reference. The programmer has intentionally used an isomorphism, a one-to-one relation between the symbols and the world to which these symbols refer, preserving the relations existing between the elements in both domains, and translated the referents' actions into a set of algorithmic operations. The relation between these domains that was once required to find the isomorphism has now become irrelevant.

The computational definition of symbols should be seen in the light of the sometimes arduous struggle to find a new isomorphism. For complex domains, it can be rather difficult to find a system of relations that works correctly. But, once the rules are designed correctly, applying them is relatively straightforward: all it requires is repeatedly recognizing a symbol, followed by the execution of an explicit, unambiguous rule. Compared to the computer's task of symbol interpretation, finding rules for the production and manipulation of symbols is by far the more difficult part of computational symbol processing.

Interestingly, the reverse is true for the social sciences. The emphasis is not so much on sign production and manipulation of symbols, but rather on the importance of interpretation. Take for instance an anthropological study of a tribe. One might find culture-specific rituals and tokens whose meaning is only understood by members of the tribe. The references of these objects and events are unlikely to be found by a superficial investigation, as it takes considerable effort to learn to understand these symbols. The symbols are often part of a complex set of conventions, whose relations are interdependent on each other. To understand the meaning of a particular tribal ritual requires knowledge of the social conventions of that tribe, much like the interpretation of a word requires knowledge of language conventions. These conventions allow the interpreter to translate a system of relations from one domain to another. In the case of a sentence, the words are translated from the domain of language to the domain of objects. As the symbol and referent are part of different domains, their connection is indirect, thus the interpreter is required to have a good grasp of both domains in order to discover this relation.

The difference between the two definitions is subtle. One might conclude that the computational definition of a symbol is a somewhat stripped down version of the humanities' definition. In the former, the interpreter may lack any knowledge of the domain that is referred to, while still being able to produce an adequate response to a signal; in the latter, this domain specific knowledge is an essential part of the sign. In order to further elaborate on the difference, a theoretical foundation for the continued study of these signs will be introduced in the next section.

Semiotics

Although signs have been studied since the earliest manifestations of philosophy, the study of signs started to gain serious attention near the end of the 19th century. One of the founding fathers of the field is the American philosopher Charles Sanders Peirce (1839-1914) who, among many other accomplishments, formulated the logic of pragmatism. He coined the term *semiotics* to denote the study of signs and sign interpretation, a term that is still used today (Chandler, 2002). Peirce's theory about the process of interpretation, or *semiosis*, covers both the interpretation of natural signs, such as medical symptoms, as well as conventional signs, which were intentionally designed to convey a message. As symbols are a particular type of sign, they are studied extensively in semiotics. A sign's meaning is relative to whoever interprets it; one may therefore say that symbols stand for something *to someone*. Peirce stresses the importance of the interpreter – human, animal or any other system which interprets signs – making it a most relevant theory for intelligent systems research. Whereas some would characterize a sign interpretation as

X is a sign of Y

Peirce includes the interpreter in the expression (Hookway, 1985):

Z interprets X as a sign of Y

So, whether someone interprets a sentence as a meaningful expression, a sequence of unknown words, or a set of ink marks on a piece of paper will depend on his or her knowledge of the common social rules and language conventions. A sign's meaning may vary among different interpreters. A person lacking sufficient knowledge or a computer program without the appropriate rules to handle input, are both unable to interpret messages as they were intended to be interpreted. Also, symbols are not necessarily linguistic signs. A logo such as the Red Cross refers to an organization by means of convention. Only those who know about this convention will be able to decode the message, others will merely notice the depiction of a red colored cross: the meaning of a sign is in the eye of the beholder. Therefore, any theory that intends to properly define symbols must explain how their interpretation depends on the capabilities of the interpreter.

To understand Peirce's view on interpretation, let's consider the elements that play a role in the interpretation process. For example, take a wolf that, while roaming through the forest, encounters a rabbit in the bushes. The wolf, in this case, is the *interpreter*. The rabbit would be what the interpretation is about, which Peirce calls the *referent*. Although the wolf perceives the rabbit by means of its eyes, it is not the rabbit *itself* that works its way directly into the mind of the wolf. Rather, it is the light reflecting of the rabbit into the wolf's eyes that bridges the gap between the two animals. The light acts as a stimulus pattern, or *signal*, between referent and interpreter.

The signal, after being affected by the referent, causes a specific reaction in the interpreting system (Peirce, 1955). This so-called *interpretant* is the instantiation of the interpretation process for a particular signal and referent. Note that there is a difference between an interpreter and an interpretant. One might compare the interpreter to a pool of water, in which a pebble – the signal – is thrown. The interpretant, then, is the specific wave pattern caused by the pebble. If we take the analogy even further, we might say that the pattern may reveal information about the pebble itself or the person throwing it. If we take a person to be an interpreter, then we would call the interpretant a thought.

With this third element, the Peircean sign is complete. The triadic sign is constituted by the inseparable union of all three elements[1]. It is schematically depicted in Figure 1.

[1] In the literature, other terms are occasionally used to denote the three elements. *Signal* is sometimes called *sign, signifier, stimulus pattern* or *representamen*. Instead of *referent*, the words *signified* or *object* may be used. The term *interpretant* is now and then replaced by *interpreter*.

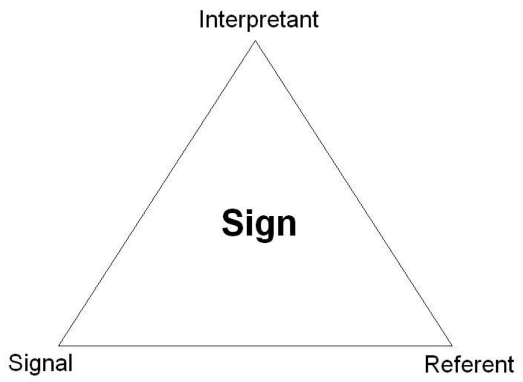


Figure 1 *The Peircean sign, depicted as a triangle (Ogden & Richards, 1923)*

The features of the three elements together determine the type of sign. Since there are many possible features for each element and each combination leads to a different type, it would be pointless to mention them all[2]. As we are mainly concerned with semiosis for the purpose of understanding the processes that take part in the interpretation of symbols, we will focus on the most important distinction: how the signal refers to a referent through a particular kind of interpretant.

Three Kinds of Signs

Before Peirce, other philosophers such as Kant and Hegel had already established that there were three possible ways in which concepts could be associated with each other. Concepts could either be similar to each other, be correlated, or be bound by a certain convention. It is likely that Peirce was influenced by this idea, as he used the same three sign types in his own categorization of signs: a signal is associated with a referent by the interpreter in one of three possible ways, which he named *icon*, *index* and *symbol*. These denotations have been used by others in various contexts, which is often a cause for confusion. However, in this thesis we will constrain their use to the Peircean definitions, described below.

The first type of sign, called *icon*, refers to an object by means of resemblance. Whenever the interpretant associates a signal with a referent based on their alikeness, the sign is an icon. A portrait, a small picture of a printer on a computer desktop, the gestures of a pantomime, a no-smoking sign displaying a cigarette covered by a red cross, an x-ray diagram, a police officer holding up his hand as a stop sign: they are all icons, because the signals share a similarity with the things they refer to. However, note that two things are not icons of each other just because they are similar: similarity, too, is in the eye of the beholder. So, whether

[2] Peirce initially contrived 10 different types of signs, but as he later found more possible dimensions their number grew to a formidable 59049 types.

two things are iconic is not an objective standard, but depends on the interpreter. In fact, if the interpreter's standards of resemblance are low enough, anything may be considered an icon of anything else.

Peirce uses the term *index* for his second type of sign. In the case of an indexical sign a signal is associated with a referent, not because they share similar features, but because the two are somehow correlated. The interpreter has previously noticed a connection between the two; now, when one of them appears it is reminded of the other. The interpretant forms the connection between the two, correlating two objects or events either in space or time. In either case, there might be a causal connection between the two or they might have a common origin. For instance, a thermometer is causally connected with temperature because it will always reach a higher marker when the temperature rises. An interpreting system that has noticed this correlation will use the thermometer mark as an index for the temperature. The signal of the height of quicksilver in the cylinder causes an interpretant that refers to the sensation of a particular temperature. Other examples of indices include windsocks, medical symptoms, footprints, smoke or the sound of a doorbell.

The third and final type of sign is perhaps the most complex: a *symbol*. Tracing the word *symbol* back to its etymological roots[3], Peirce translates it to "a thing thrown together" (Peirce, 1894). The Greeks used this word as a figure of speech to signify a contract or convention, or in more general terms *a signal that is agreed upon*. The essence of symbols is that they have been established by social convention. Unlike an icon, the connection between signal and referent is not based on resemblance. Unlike an index, a correlation between the two is not required. Rather, they are connected by the interpreter itself, in a seemingly arbitrary way. Any signal may cause an interpretant that refers to any referent, without the necessity of a natural link – either similarity or causality - between the two. Words are typical examples of symbols. There is nothing about the six letters of the word *rocket* that hint towards its meaning. The number of occurrences of the word together with an actual rocket is usually not high enough for them to be correlated. There may be a link between signal and referent in the brain, but there is none between the two in the world. Instead, both communicating parties have an agreement on what a word means, and the interpreter assumes this agreement still holds when the signal is interpreted.

Because a symbol's reference is determined by convention, meaning is not intrinsic to the physical appearance of the signal. Typical examples of symbols are: a wedding ring, a red cross, a number or a flag. Another way of thinking about symbols would be to view them as metaphors. In a sentence like

> *"A tree stands to a squirrel like a house stands to a person"*

there is a metaphorical connection between trees and a houses. Trees, squirrels, houses and humans do not share any striking similarities, nor is a tree usually associated with a house

[3] The Greek word σύμβολον (symbol) is composed of the words σύμ (sym) which translates to "together" and βολή (bole) meaning "throw".

based on some remarkably rare correlation. Using a word to stand for something else is typically based on convention, making the usage of the word "tree" *as though* it is a house something other than a mere pointing relation.

When considering the examples given, one might wrongfully conclude that all of these are necessarily icons, indices or symbols because of some kind of intrinsic property. However, they are only signs of a particular kind insofar as they are interpreted to be so. Signals are not intrinsically icons, indices or symbols, because the type of sign depends on the interpreter's capability to produce the required interpretants. As in the example of a red cross, it depends on the interpreter whether something is an icon or a symbol. In fact, signs are often both icons and symbols[4] at the same time, such as the word *hiccup*: it is an icon because its pronunciation sounds like its referent, but it is also a symbol since its meaning is determined by convention. The same goes for a non-smoking sign. Someone who claims to be unaware of the specific convention of this sign may still be held responsible for smoking, as the sign is clear by itself.

What makes all these examples typical for their respective types of signs is not some kind of intrinsic property, but rather that they seem fit for a particular purpose (Deacon, 1997). Many of them were designed to relate to a referent in a certain way. A portrait should look as much as possible like the person it depicts. Pictograms are designed to be understandable for people that are unaware of the ruling conventions, for example on locations with many tourists such as airports or train terminals. The purpose of a windsock is to signify the direction of the wind, while a doorbell is designed to announce that there is a person standing in front of the door. Wedding rings are a relatively simple sign for a whole array of virtues that are generally associated with marriage, such as bonding, eternity and equality. By convention, a rectangle containing several colored stripes and dots signifies a country, its inhabitants, its customs and many other traits.

Notice how signs that were designed to be interpreted as icons consistently differ from symbols. Usually, symbols are relatively simple objects that have a very complex referent. Icons on the other hand have to be much more complex objects, since they have to be similar to their referent. This causes many constraints on the design of icons, while symbols can be selected more freely. Therefore, the production of icons, such as drawing a portrait, can be quite difficult. The design of new symbols is relatively easy: all one has to do is to decide for a signal to represent a referent. For their respective interpretation the reverse is true. Since icons resemble the thing they refer to, it takes almost no effort to understand them. A perfect icon would look just like its referent; consequently linking the two together is a trivial matter. Symbols are much harder to understand, as they require the interpreting system to be configured in such a way that a particular signal causes a particular interpretant, which refers to a particular referent. If the interpreter is unaware of this convention, the intended symbolic interpretation will fail.

[4] The general name for this class of words is *onomatopoeia.*

This is not the first time we have seen such a reversed correlation between the production of signs on the one hand, and their interpretation on the other: we have drawn the same conclusion from the computational and social science's meanings of the concept *symbol*. Icons collate with computational symbols, since both of them are hard to design and in both cases the interpretation is relatively effortless. They are interpreted on the basis of their physical similarity with their referent, the premises of a rule. The Peircean definition of a symbol is more like the social science perspective on symbols: a sign that is easily produced, but takes a considerably sophisticated system to interpret correctly. It requires a deep understanding of the network of relations into which the symbol is embedded. With this in mind, we can finally distinguish the two modes of understanding in the Chinese Room. By following the rules in the dictionary we are doing an iconic interpretation of the Chinese characters on the note, only paying attention to their form and the similarity with the premises of the rules. A Chinaman would actually interpret the characters symbolically, relating them to their intended referents.

The purpose of this shift towards the semiotic theory is not just to make the distinction between the two forms of interpretation more clear, or to label the distinction made with the Chinese Room argument. More importantly, the shift serves a practical purpose. Previous definitions of symbolic interpretation were based on external observations, such as the Turing Test, or arguments lacking scientifically unambiguous terms like Searle's different modes of understanding. Contrarily, the definitions of *icon* and *symbol* are embedded in a larger theory of interpretation. Instead of limiting ourselves to a discussion of the observable features of interpretation, we can now focus our attention to the processes that play a role *inside* the interpreter.

Chapter 3: Hierarchy of Signs

Constructing a Ladder

In his book *Mind and Nature* (Bateson, 1979) the British anthropologist Gregory Bateson introduces his epistemology by exploring the relation between interpretive processes and the patterns which exist in nature. One of the book's central themes is what he calls *the pattern that connects*. Right at the beginning of the book, he presents the reader with an analysis of the similarities and relations between the physical structures of organisms to explain this central concept:

> *The parts of a crab are connected by various patterns of bilateral symmetry, of serial homology, and so on. Let us call these patterns within the individual growing crab first-order connections. But now we look at crab and lobster and we again find connection by pattern. Call it second-order connection, or phylogenetic homology. Now we look at man or horse and find that, here again, we can see symmetries and serial homologies. When we look at the two together, we find the same cross-species sharing of pattern with a difference (phylogenetic homology). And, of course, we also find the same discarding of magnitudes in favor of shapes, patterns, and relations. In other words, as this distribution of formal resemblances is spelled out, it turns out that gross anatomy exhibits three levels or logical types of descriptive propositions:*
>
> > *1. The parts of any member of Creatura are to be compared with other parts of the same individual to give first-order connections.*
> >
> > *2. Crabs are to be compared with lobsters or men with horses to find similar relations between parts (i.e., to give second-order connections).*
> >
> > *3. The comparison between crabs and lobsters is to be compared with the comparison between man and horse to provide third-order connections.*
>
> (Bateson, 1979, p.10)

Thus, Bateson recognizes three different types of relations between objects. His first-order connection depends on a comparison of parts, a judgment based on similarity. His second-order connection requires a comparison between relations of parts of animals. And in the case of Bateson's third-order connection, the comparison of two animals *itself* is compared to another comparison. In this chapter, we will find that Peirce's sign trichotomy is not that different from Bateson's account of ordered patterns (Hui, Cashman & Deacon, 2006).

In the previous section we have demonstrated how signals are related to referents by means of three different types of interpretants. The primary goal of this section is to show how icons, indices and symbols are related *to each other*. These sign types are not equally interchangeable: as they increase in complexity from icon to symbol, they progressively rely on the more simple forms of interpretation. This hierarchy is also evident in Bateson's levels of descriptive propositions. Just as first-order connections serve as a foundation for second-order connections, and second-order connections build up to third-order connections, so do symbols depend on indices, and indices on icons. This will prove to be a crucial point in the design of the symbolic interpreter's blueprint. We should therefore take some time to understand the processes that underlie the formation of interpretants, and how they are related. In the remainder of this chapter, two examples showing different interpretation processes will be presented and analyzed following a general scheme that was previously set out by Deacon in his book *The Symbolic Species* (Deacon, 1997).

Imagine a museum where a 17th-century painting is on display, depicting a royal meal in the dining hall of a castle. The king and queen are located in the center of the canvas, and the queen is holding a dog on her lap. A child, visiting the museum with its father, notices the animal and remarks it looks just like a dog. Quite clearly, this is an iconic observation. When the father sees the dog he wonders where its doghouse might be, based on his previous experiences with dogs and doghouses being together. Using the correlation between dogs and doghouses is an indexical interpretation. The director of the museum walks by and explains what the painter's original intentions were: back in the time when the painter lived, painting a dog on someone's lap was a symbol for adultery. By painting one near the royal couple, the painter intended to mock them. This symbolic kind of interpretation is most likely to be done by an expert. It requires a deep understanding of the conventions that were common in the 17th century.[5]

Another example: say a history professor is holding an umbrella upright in front of him, while walking through a University hallway with a colleague. A student outside the building takes a peek through a window and recognizes the professor, but he can't quite distinguish the umbrella. He thinks it is a walking stick. As his judgment is based on the physical similarity between an umbrella and a walking stick, it is an iconic interpretation. Another student passing the professor in the hallway *does* recognize the umbrella and without taking a peek outside, he immediately concludes that it must be raining outside. This is an indexical interpretation, because he knows that umbrellas are often seen when it is raining. Note that he must recognize the umbrella previous to being reminded of the correlation. The indexical interpretant depends on an iconic classification.

All the while, the professor is telling a story to his colleague during their stroll. He tells how gladiators would fight in ancient Roman arenas and illustrates his story with gestures, using his umbrella for a sword. The colleague interprets the umbrella symbolically, as it stands for a sword. This differs from an iconic interpretation. He doesn't think the umbrella *is* a sword: he

[5] For a more thorough account of sign interpretation in the visual arts, cf. (Panofsky, 1972)

can easily see it isn’t. Also, he is not just reminded of a correlation between swords and umbrellas, because those are usually not seen together. Rather, he interprets the umbrella *as though* it is a sword. How did the history professor arrange this convention? The physical features of an umbrella alone are not sufficient to remind his colleague of a sword. He would never guess what it is supposed to stand for without the conventions set up by the professor telling his story. These conventions are created by highlighting the relations that the two objects have in common, or by inventing new ones. For instance, by swinging the umbrella like a sword the professor intends to show a correlation that both objects have in common. Or by holding it upright in front of his chest, he shows another resemblance. It is important to note that this is not the usual kind of physical resemblance: it is a likeness between *relations* of two objects. The story creates a similarity in the topologies of both *networks of relations*, a similarity that may be recognized by a symbolic interpreter.

Both examples show how each type of interpretation requires an increasing competence. They may also unveil how each type of interpretation depends upon the others: each type relies on the capability of interpreting a more simple type. Iconic interpretation skills are required for an indexical interpreter; a symbolic interpreter needs to have indexical skills. In order to discover the kind of interpretant that are responsible for this hierarchy, we will reconstruct the two examples from an interpreter perspective, starting with the formation of sub-symbolic – iconic and indexical – interpretants.

Sub-symbolic Interpretation

For the student who is staring at the umbrella from far away, thinking it is a walking stick, it makes no difference whether the professor is actually holding a walking stick or an umbrella: he will believe it is a walking stick anyway. The child, not having seen too many dogs in its life, would not have noticed the difference between two different dog breeds, and still classify the creature as *a dog*. For someone who is blind, all paintings are icons of each other. In each of these examples the interpreter typically fails to make a distinction. Again, although things that have similar physical features will often tend to be regarded as iconic of one another, it is the particular interpretant caused by these features that determines the sign type. It is likely that some differences will be disregarded when the signal causes an interpretant. Therefore, iconic interpretation basically comes down to a process of *classification*.

Iconic interpretation is the necessary starting point of every interpretation process – see Figure 2. In the first example, both the father and the expert have to recognize the dog, before they can make an indexical or symbolic interpretation, respectively. The same goes for the example with the professor´s students and colleague, who have to recognize the umbrella before discovering any indexical link or symbolic meaning.

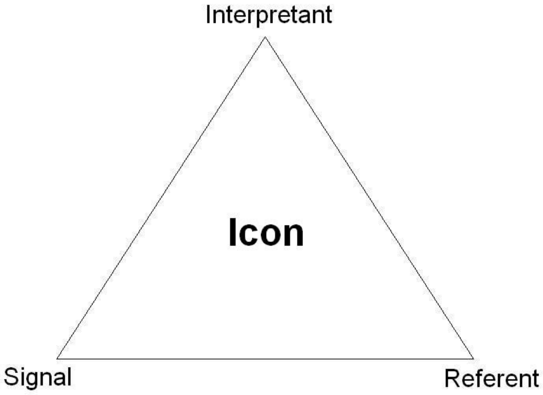


Figure 2 *The iconic sign*

Like icons, indices only exist by virtue of the interpreter noticing a correlation in space or time. There are no objective conditions to the proximity of two things, or the time interval in which they occur that makes them indexical. Only the standards of the interpreter determine whether they are an index, although close proximity or simultaneous appearance *do* make an indexical interpretation more likely. An index supposes a correlation between two things, based on previous experiences of those things occurring together. In the first example, when the father of the child recognizes a dog on the painting he is reminded of the correlation between dogs and doghouses, two concepts which can often be seen together. The iconic interpretation of the dog is primary: the visual stimulus causes an interpretant, classifying it as a dog. But now, the interpretation process continues and the first interpretant causes another interpretant by virtue of the previously noticed correlation. In this case, the second interpretant refers to a doghouse. It is the same iconic interpretant that would be caused by the visual stimulus of a doghouse, but now it is caused by another interpretant. The image of a dog brings the thought of a doghouse to the father’s mind. The same goes for the student passing the professor. He sees an umbrella, which makes him think about rain.

In the examples we have seen how icons bring about other interpretants, by virtue of a correlation between the two. However, the classification of the visual stimulus is not the only icon taking part in this process. The father has seen doghouses before and noticed they all shared a similarity too. Now, this iconic interpretant is caused again, albeit this time by another interpretant. The third part of the index is formed by a correlation between dogs and doghouses. Since this particular pairing bears a resemblance with previous pairings between dogs and doghouses, it is a *higher-order icon*. The index only works because the interpreter has repeatedly recognized that these two icons appear together, both now and previously. In many situations there has been a correlation between a dog and a doghouse. This particular correlation is iconic to these. Thus, three similarities ultimately play a role in the indexical interpretation process: a similarity between the dog and previous occurrences of dogs; a similarity between the doghouse and previous occurrences of doghouses; and a higher-order similarity between the co-occurrences of dogs and doghouses. An index depends on icons while icons themselves do not require indices; therefore they are hierarchically related, as depicted in Figure 3.

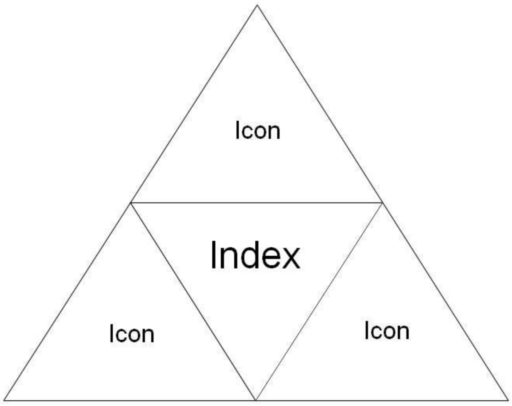


Figure 3 *An index, composed of three icons*

Symbolic Interpretation

It should be no surprise that we find symbols at the top of the sign hierarchy. Explaining symbolic interpretation requires a slightly more thorough investigation of the processes involved than that was required for our analysis of iconic and indexical interpretation. We have already seen that a symbol is a complex kind of sign. Its physical features are not necessarily intricate but its interpretation usually is, as it requires knowledge of the conventions surrounding it. The museum director is the only person who has enough expertise with these paintings to be aware of the convention that dogs used to be associated with adultery. There is no natural correlation between the two, neither in time nor in space, and so it is not immediately clear why they should be associated with each other. Their relation appears to be arbitrary; the convention of the metaphoric use of a dog's depiction has simply been passed on over time. Someone, at some point, found a reason why the two should be connected and shared this rule with others. Narrowing down our analysis, we find three important questions that beg for an answer. In what way are conventions communicated generally? What reason could someone have to link two things together that are not directly related by similarity or association in the first place? And what kind of interpretant plays a role in the construction of such a convention?

In the example of the professor using his umbrella as a sword, he himself already knows what the umbrella stands for. In order for his colleague to understand the metaphor, he intends to highlight the parallel between the two different settings in which umbrellas and swords are used by holding the umbrella in specific positions, swinging it around and using specific gestures and facial expressions. Both objects are embedded in a network of indexical relations, of which the professor's activities are also a part. A sword can remind someone of a swinging object in general, or an object that is held upright in front of the chest. By showing these particular motions, the professor adds indexical relations to an umbrella that match with some of the indices a sword has, thereby aligning their indexical networks. At some point, the colleague may notice the similarity between the two networks and remark that this umbrella is just like a sword, because they have so many similar indexical relations. The discovery of this metaphor is exactly what the professor is aiming for. The goal of his actions is to align the networks to increase the probability of the students being reminded of a sword, even though they can see it is an umbrella.

But how does the professor himself link these objects? After all, he is not induced by someone else to see the similarity of the indexical relations, but has to notice it on his own. Without another person swinging around an umbrella standing next to him, the network of indices in which an umbrella is embedded will probably be very different from the network of a sword. However, in our discussion of the iconic interpretation process we have seen that whether two things are alike is not an objective measure, but follows from the standards of the interpreter. When distinctions are ignored, objects with very different physical features can still be regarded to be the same. And just as anything can be iconic of anything else, so can the topology of any network of relations be iconic to any other network's topology. If two things share many similar indices, they are more likely to be regarded as symbolic of each other, but two things that have entirely different relations may still be considered symbolic of each other.

In more general terms then, a symbol requires a *system iconicity* between the topologies of the networks of indices in which the symbol and its referent are embedded. A *higher-order indexical* relation between two objects or events is formed, based on the likeness of their respective indexical relations. Whereas an index links two things together that co-occur, a symbolic sign points to something by virtue of their common locus in comparable network topologies. As the example of the umbrella and the sword demonstrated, the resulting relation is not just a correlation but a higher-order index based on other indices. In this relation, a symbol token stands for an action or object, provided that their indexical relations are properly aligned for the interpreter to notice the system iconicity.

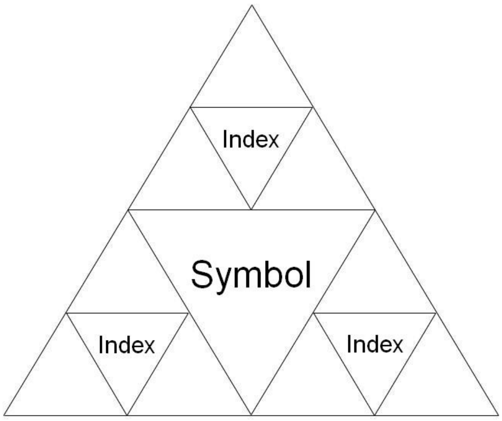


Figure 4 *The Peircean symbol, hierarchically composed of three indices*

The diagram shown in Figure 4 may be misleading in its simplicity. One might conclude from it that a fourth type of sign could easily be contrived by adding another set of triangles. However, if we consider in what ways an interpretant can connect a signal to a referent, it becomes clear that there are only three types. The most basic unit is the icon, followed by the index. The third type is formed by a similarity between the indexical network topologies of signal and referent. If we suppose there are more than three types possible, the connection between signal and referent for this fourth type of sign would be based on a higher-order index among higher-order indices among the indexical network topologies, yielding yet another symbol.

We have seen in this chapter how interpretations can occur in various ways, depending on the properties of the interpreting system. It turns out that both iconic and indexical qualities form a necessary component of a symbolic interpreter. The hierarchical relations binding these three levels are a crucial part of the theory set out in this thesis. However, at this point our model is still quite formal, lacking any evident applications. In working towards a falsifiable model of our theoretical system, we will investigate the possibilities for a more concrete implementation of these ideas in the next chapter.

Chapter 4: Symbols and Networks

Neural Networks

Leaving the hierarchical ordering of signs aside for a moment, let us focus on the vehicle of interpretation and specify what characteristics define an interpreter. Peirce himself adheres to a rather broad definition: any system could principally be regarded as an interpreter, provided that its properties satisfy the necessary conditions for interpretation (Hookway, 1985). That is, such a system should be capable of producing an interpretant that makes a connection between the signal and the referent of the sign, either by similarity, correlation or convention. To determine what kind of interpreter a system is, it needs to be analyzed based on its potential to produce interpretants of a certain type. So whether living or non-living, a human brain, wolf brain or rabbit brain, computer, glass of water, or artificial neural network: any system could principally be regarded as an interpreter. But although these are all potential interpreting systems and therefore potential vehicles for our theoretical model, they are not all equally interesting in the light of this research. Specifically, they differ in degrees of comprehensibility, biological plausibility and the type of interpretants they can produce.

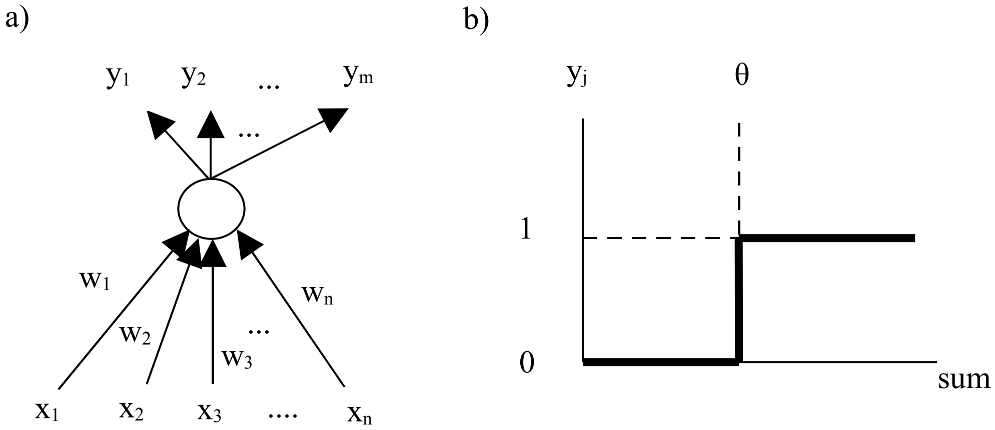
Considering these factors and the possible systems, artificial neural networks appear to be a good candidate. Being a mathematical abstraction of a biological process, they are modeled after the electronic pulses propagating neurons in the brain. We know that our human brains are capable of symbolic interpretation, which, due to the hierarchical nature of interpretation, implicitly implies a capability for indexical and iconic interpretation. Because artificial neural networks are based on brain processes, it could be argued that they also meet the necessary conditions for producing symbolic interpretants. Although this is only an assumption, their biological plausibility gives them a head start when it comes to finding the most appropriate model. Additionally, neural networks are regularly used in artificial intelligence research. Their generic nature allows for an array of different learning tasks, such as automatic recognition and prediction. They are also widely used in symbol grounding studies (Harnad, 1990), (Vogt, 2002).

Neural networks are sometimes criticized for being incomprehensible to human observers. Unlike the traditional rule-based systems where the execution of a program can be traced and, to some extent, understood, neural networks are generally treated as black boxes. Generally, only their input and output values are considered while the firing units in the hidden layers are ignored. However, when we consider this apparent weakness from a different perspective, it turns into an advantage. The human observer who makes sense of a program's execution runs the risk of attributing properties to the computer that are unaccounted for, because variables and functions are usually named for what they *ought* to do, but not always for what they actually do[6]. Using a variable called *learning* in a program might falsely induce someone to think it is learning. A neural network, or any other black box model, is less likely to be

[6] This error is also known as the fallacy of *wishful mnemonics* (McDermott, 1981)

attributed capabilities unjustly, because its components have no obvious meaning to a human observer. This necessitates a careful examination of the network's interpretation processes to determine what is actually going on inside the system, and how or why an interpretant was caused.

Let's take a closer look at neural networks and how they work, in order to see how they can be mapped onto Peirce's semiotics. A typical neural network is composed of several *nodes*, sometimes called *processing units*, interconnected by a number of edges. Inspired by biological neural networks - where neurons become activated by an electrochemical pulse and subsequently fire a pulse to other neurons through synapses – nodes are activated by an *activation function* that operates on its incoming edges. Once activated, the node in its turn activates other nodes through its outgoing connections. Figure 5 shows an example of a node and a common activation function.



$$\mathrm{sum} = \sum_{i=1}^{n} (w_i x_i)$$

$$y_j = \begin{cases} 1 & \text{if sum} \geq \theta \\ 0 & \text{if sum} < \theta \end{cases}$$

Figure 5 *(a) A neural network node, with an activation function y over the weighted sum of its input connections. The activation of each input x_i is multiplied by a weight variable w_i for that vector. (b) If their summation passes a certain threshold θ, the node fires by activating its outgoing connections y_j. In this example, the* step *activation function is used to determine the output value of the node (McCulloch & Pitts, 1943).*

By varying the number of nodes and the configuration of connections, different networks with different abilities can be realized, making them popular for tasks involving learning. In general terms, learning occurs by generating a large amount of different network architectures, and holding on to those which cause the closest approximation of the required output values. Since the number of possible random architectures will generally exceed the number of models that can be tested in a reasonable amount of time, a heuristic method is commonly used to generate new architectures. Two prevailing methods for exploring the search space are the Backpropagation algorithm (Rumelhart, 1986) and Genetic Algorithms (Holland, 1975).

It can be quite difficult to follow what goes on in these clusters of interconnected processing units, especially when the number of nodes grows larger. To keep neural networks somewhat comprehensible, the nodes are typically[7] grouped in three kinds of layers: an *input layer* where data enters the network; several hidden layers that propagate the activation pulses; and an *output layer* that shows the result of the propagation to a human observer or a learning algorithm. Note that difficulties arise whenever one ascribes interpretation qualities to the network. For instance, are the output neurons a vital component of the interpretation process, or do they just serve as indicators to an external observer? How is the network's knowledge distributed over the system? Are certain neurons responsible for particular tasks, or is every neuron involved each time? And how do we distinguish different types of networks based on their structure? Although these question fall outside the scope of this thesis, they beg for a theoretic framework of interpreting systems into which neural network models can be embedded.

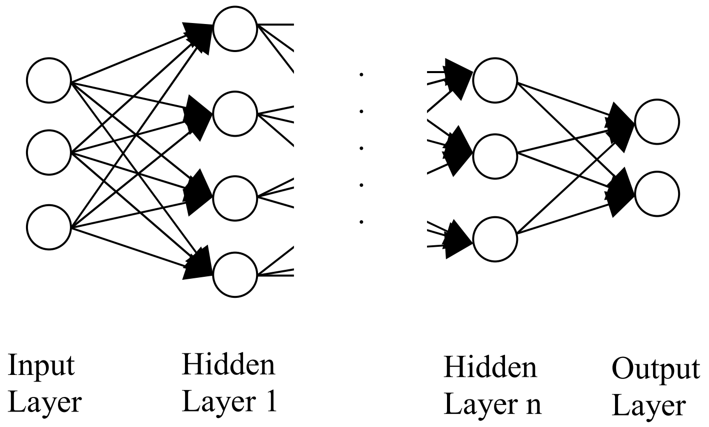


Figure 7 *Example of a feed-forward neural network. Data enters the network at the input layer on the left. After a series of parallel propagations through the hidden layers, the output nodes are activated.*

[7] This claim is true for most *feed-forward neural networks,* the type of networks used throughout this thesis. Other types (e.g. *recurrent neural networks*) often have differently named layers or a completely different architecture altogether.

Interpreting Networks

Let's reconsider the semiotic theory of Peirce, which has been discussed in chapters two and three, and see how it applies to neural networks. As the Peircean sign consists of three parts – signal, referent and interpreter – we should be able to distinguish all three in the neural network interpretation process. We will do so by analyzing an example where a neural network classifies a series of images. We construct a training set containing various visual representations of circles and squares, which a neural network learns to classify into two different categories. Each time an image is shown to the network, a sign interpretation takes place.

Clearly, the images should be considered as *signals* to the neural net, which itself is an *interpreter*. The propagation of pulses through subsequent layers of the network – a particular activity pattern caused by the signal – is the *interpretant*. Finally, the nodes in the output layer indicate whether the signal was found to refer to a circle or a square. Because the network does not have direct access to these *referents*, it attempts to reconstruct them from the information available in the signal. The network could also have been taught to separate small from large objects, or blank images from random markings. But in the case of this example, the reconstruction process only takes salient properties into account that discern circles from squares, as the network has learned to distinguish between these two particular shapes.

Note how the semiotic approach leads to a limited but functional understanding of what goes on in these networks. The fallacy of wishful mnemonics is carefully avoided: we should not let our own understanding of these concepts get in the way of theorizing about the network's interpretation capabilities. No claims are made about the similarities between our conception of circles and those of a neural network - we can safely assume that the differences are plentiful. Instead, it is only claimed that, for this particular network, a circle is defined by its properties that make it distinguishable from squares, and vice versa. In other words: to this network, every input is either a circle or a square. If the network had been trained differently, by for instance rewarding the classification of a third category of triangles[8], we would have affected the possible set of referents and thereby the network's conception of circles and squares.

As a next step towards understanding the interpretation process that takes place in the example above, let's contemplate on the type of interpretation at hand. Is the classification of circles and squares an iconic, indexical or symbolic task? In the case of iconic interpretation, the signal is connected to a referent based on a similarity relation. The interpretant caused by the input signal generally leads to a prediction of the referent based on the typical characteristics of the image. In this case, the relation between signal and referent is indeed

[8] Remark that the training set does not necessarily have to include images of triangles. We merely train the network to classify some images as triangles based on their similitude to our conception of triangles. It is important not to confuse the shapes that correspond to words like circle, square or triangle to us, with the neural network's conception of these categories.

based on similarity, or rather, the network's ability to discriminate between various shapes. Differences within each of these categories are discarded in favor of properties that allow for a separation between circles and squares.

It is generally known that neural networks are well fit for these kinds of classification tasks. Their robustness allows them to handle distorted data, while their generic architecture facilitates many different learning algorithms, even allowing for classification without feedback (Kohonen, 1982). Not coincidentally, Harnad uses the term *iconic representations* when he discusses how a neural network is used as a basic building block for connecting intelligent systems to the environment (Harnad, 1990). A neural network is indeed a suitable model for an iconic interpreter. But how about indexical interpreters? Can we still use neural networks as a model for the kind of interpreting system that draws a link between signal and referent by means of correlation?

Fortunately, we can. Another common task for neural networks, apart from classification, is prediction. By showing it a number of input-output pairs, the network can be trained to predict these correlated pairs by activating the output state when the input data is shown. We can compare this to the example in the previous chapter, where the recognition of a dog brings the thought of a doghouse to the mind of the observer, without him seeing an actual doghouse. When the input data is shown to the network, it causes an activation pattern as though the correlated data had been shown directly. And, like in the example, this indexical relation consists of three icons: the activation pattern caused by the input data, the pattern that would have been caused by the output data, and finally the higher-order pattern of the network that causes the first to turn into the former. All three of these responses have to be learned correctly by the network in order to properly execute an indexical interpretation.

Consider the task of diagnosing a patient based on medical symptoms. A neural network has learned to predict the patient's ailment knowing only these indicative symptoms. It has repeatedly been shown a training example, after which it received feedback on the correlated disorder. In this example of indexical sign interpretation, the signal is the set of symptoms. As Peirce suggests, an interpretant can be a signal for yet another sign; we see here that the symptoms, being themselves iconic interpretants, are a signal to another, indexical interpretation process. The specific disorder is the referent of the sign, while the network's habit of going from one icon to a correlated icon – a higher-order activity pattern - is the interpretant of this indexical interpretation. In doing so, the interpretant relates the signal to the referent because it has previously observed their correlated presence.

One way of modeling an indexical interpreter can be achieved by using so called *recurrent neural networks* (Elman, 1990). These systems have the property that some layers are bi-directionally connected, creating loops in the network's architecture and thereby adding a time dimension to its activity. Elman shows how this network can learn to associate words that appear next to each other in a set of common sentences. These words are presented one at a time, leaving the network to discover a correlation in time between them. They are primarily interpreted as icons, but at the same time they also give an indication of which word will be next. For instance, one might expect the word *Hong* to be followed by *Kong* if one is familiar with the name of the territory but not with the Chinese language. In the same way, when the recurrent network recognizes the word *Hong*, it has the habit of causing a certain activity pattern that would normally be active when the word *Kong* is recognized. Because of this higher-order tendency that it has acquired during the learning stage, the system is able to predict what the next word will be.

As the loops in recurrent neural networks add a lot to their complexity, these systems have proved to be quite difficult to use, let alone to understand their workings in the light of Peircean semiotics. There are, however, models of lesser complexity that still meet the preconditions for indexical interpretation. The same neural network that we used for classifying circles and squares can also serve as a model for diagnosing symptoms. If we take the symptoms to be the signal for the input layer, and the diagnosis as the referent in the output layer, the network can be trained to produce interpretants that connect symptoms with their corresponding disorders. The network thus learns to make an indexical mapping from one icon to another. Both icons consist of rather trivial classifications from an input signal to an identical referent in a network consisting of one node. But keeping the icons that simple consequently allows for the indexical interpretation to be quite straightforward. Despite the superficiality of the classification process, this model is still preferable over other models like recurrent networks due to its simplicity. And although several other issues can be raised about the self-organizational abilities and the biological plausibility of this system[9], it serves as an adequate model for indexical interpretation considering our purpose of discovering the architecture of a symbolic interpreter.

Emerging Symbols

The final question that we need to answer is how this indexical neural network leads to a model of symbolic interpretation. After all, the semiotic hierarchy that was covered in chapter three implies that symbols can be constructed from indices just as indices are constructed from icons. One possible approach would be to apply the same trick that was also discussed in the previous section; only this time, we claim that both the iconic *and* the indexical step – recognition and subsequent association of the input data - consist of a trivial transition within one node, resulting in a network that makes a 'semantic' mapping between patterns. For instance, we could construct a simple neural network that connects a pattern that we have named *umbrella* with another pattern we call *sword*, and claim that the network understands that umbrellas can be used as metaphors for swords.

However, just because *we* sometimes use an umbrella as a symbol for a sword, it doesn't automatically follow that every device connecting these two concepts is also symbolic. If we take the same network and patterns, but now call the patterns *dog* and *doghouse*, the network doesn't suddenly become an indexical interpreter. Our own interpretations should not stand in

[9] What these issues are and how they affect the scope of this theory will be discussed in the final chapter.

the way of trying to make an objective assessment of the network's capabilities. This line of argument bears a resemblance to Searle's critique on rule-based systems. He intends to show that a mapping that would normally require a symbolic interpreter – answering a question in Chinese – can also be done with an indexical system – a simple, rule-based dictionary (Deacon, 1997). In the same sense: a mapping of *umbrella* to *sword*, or a word to a corresponding concept, is not always a symbolic operation. Peirce shows that semantics, the symbolic connection between signal and referent can only be achieved using an interpreter with symbolic qualities. Therefore, we will approach the question by considering how such a symbolic interpreter can be constructed from indexical systems, using the Peircean hierarchy.

Recall the three binding factors of signs: similarity, association and convention. The latter is no doubt the most difficult type of sign, often leading to ambiguous definitions. How do we represent convention in a neural network? Classification by similarity takes care of icons, and we have shown that both recurrent and simple, feed-forward neural networks can be used for modeling indexical interpreters. As the previous arguments show, another feed-forward network can theoretically be used to model symbolic interpretation by constructing a mapping between concepts we would consider as metaphors. However, such a superficial model does not lead to any interesting insights into the nature of metaphors, nor does it demonstrate what qualities a symbolic system should possess. We should delve deeper into the workings of symbolic interpretation in order to come up with a more realistic and informative model.

Recall the example of the professor swinging and stabbing with his umbrella to remind his colleague of a sword. It has been argued that this intentionally induced symbolic connection consists of three parts:

> *(1) The indexical network of relations in which umbrellas are embedded;*
>
> *(2) The indexical network of relations in which swords are embedded; and*
>
> *(3) A higher-order link between these two networks.*

Both (1) and (2) are presumed to be available to the colleague, but initially they are not similar enough for (3) to be noticed. By adding indexical links like swinging and stabbing, the professor intends to expand (1) in order to make it more similar to (2). His goal is to make the colleague notice that (1) and (2) are iconic of each other. Once the similarity is noticed and their relation understood (3) allows the interpreter to pick a token of one system - an umbrella - and use it as a symbol for one of the other system's tokens – a sword in our example.

Keeping this example in mind, how would we model such a hierarchical ordering of interpretation? We have already seen how a simple feed-forward neural network can be used for indexical interpretation. This means that (1) and (2) should each be represented by a neural network, in which the concept's relations are represented by an associative mapping. Using a truth-table containing the correct input and output patterns, the network can be trained to associate one icon with another. This leaves one final question to be answered: how can (3), the higher-order connection between these networks, be established?

To understand the metaphoric use of an umbrella for a sword, a similarity beyond the pure physicality of the objects is required; instead of a comparison of the qualitative aspects of these objects, the professor´s co-worker now has to compare the indexical relations. Notice how this juxtaposition is *itself* an iconic interpretation, this time using the interpretants of both networks as a signal for a higher-order interpretation. Therefore, in order to find the similarity, any iconic interpreter may be used – including a neural network!

Once this similarity has been found, the symbolic system uses the redundancies among both indexical networks in order to learn more efficiently. Not only is the higher-order layer responsible for finding a *metapattern* - a pattern of patterns - among these two domains, it should also relate them by establishing a higher-order index, as the symbolic link is constituted of a pointing relation between two indexical interpretants. The *metalayer* needs to interpret the interpretants, recognize the similarities in their correspondence relationships and relate concepts in similar positions. And, since neural networks are also capable of indexical interpretation, we can use them to model not only the iconic process of this metalayer, but also the higher-order index. This leaves a hierarchical ordering of three indexical neural networks to model a symbolic interpreter: two networks representing both domains and a third to observe and relate the other two.

This also explains why we have consistently called the latter a higher-order network: because its signal is formed by the interpretant of yet another network. By gaining *insight* into the processes of the other two networks, this third network is able to draw a higher-order link between the two, allowing the system to associate two concepts metaphorically – that is, not by correlation but by convention. We conclude this chapter by quoting, again, Bateson on the hierarchy of interpretation:

> *We have constructed a ladder [...] The pattern which connects is a metapattern. It is a pattern of patterns. It is that metapattern which defines the vast generalization that, indeed, it is patterns which connect.*
>
> (Bateson, 1979, p.10)

Chapter 5: Symbolization, Language and Neural Nets

The Chimpanzee Experiment

To evaluate the merits of the model presented in the previous chapter, the difference between indexical and symbolic interpretation should be made more concrete and used in an experimental environment. However, finding such an environment poses a serious challenge. We have discussed in earlier chapters how difficult it can be to compare artificial models of representation with natural models. Recall that one of the strengths of the Turing Test is that it skips this problem entirely and focuses exclusively on the externally observable difference between human and computational inference. But as the Chinese Room Argument demonstrates, we can't just conclude that a program's computation is symbolic merely from the correctness of its output sentences. For such a conclusion to be drawn, we need a proper understanding of how the output comes about. Perhaps the most convenient solution – albeit an impossible one - would be to look directly inside a computer, study every piece of a computational interpretation process and compare these to human brain processes. Unfortunately, the incompatibility of these system's architectures in size, structure, and many more dimensions, forces us to take a less direct approach.

Despite the limitation of only having indirect access to the object of our study, the interpretation process, much information can still be gathered from experimental data – indeed, more than Searle's argument implicitly suggests. It may not be valid to ascribe symbolic qualities to a system that merely produces a correct sentence; but we will see that by comparing learning strategies and experimenting with different kinds of learning tasks, such sentences provide a world of knowledge, given an adequate theoretical foundation.

To demonstrate how analyzing output sentences can lead to conclusions about the semiotic properties of a system, we turn to a series of language training tasks for chimpanzees. In the 1970's, Sue and Dwight Savage-Rumbaugh devised and conducted several experiments to test the linguistic capabilities of these apes; an overview of their work is presented in *Symbolization, Language and Chimpanzees* (Savage-Rumbaugh, 1978) in which they relate the experimental results to the Peircean framework. They were able to show how different language acquisition strategies explain the varying learning curves between apes, using Peirce's distinction between icons, indices and symbols to interpret their data.

Previous to that article, other language training studies had already claimed that apes could learn a vocabulary of over a thousand words. As apes are not able to express themselves clearly enough by vocal communication, a panel containing small pictures called *lexigrams*, shown in Figure 8, is typically used in these kinds of experiments. An ape has to press or point to one of the lexigrams with its finger, thereby indicating the intended use of that word and demonstrating its language capability. For example, in order to receive a banana, he might point to the lexigram associated with *banana*, and – depending on the complexity of the learning task at hand – subsequently point to the *give* lexigram.

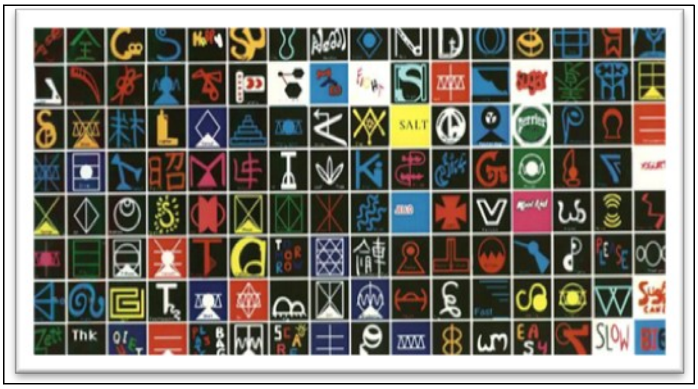


Figure 8 *A typical set of lexigrams. Some of these are composed of basic elements such as lines and circles; others depict images that we would find iconic of objects or people. The particular lexigrams depicted in this image are a part of the* Yerkish *language, a set of tokens specifically designed for ape communication.*

Using the training methods such as the one described above, apes can be induced to learn large vocabularies. But as the number of studies grew and the gathering of experimental data continued, some researchers started to doubt the validity of the claim that these apes were using language. Surely, the sentences were produced without error. And given the limited set of words and sentences, their behavior could certainly not be discerned from humans performing the same task. However, it was disputed whether the sentences were really symbolic, or just the result of a stimulus-response correlation that was learned by the apes. The lexigrams they pointed to might not *mean* anything to them; the pointing could simply be an action learned to obtain a reward.

Critics of these simple naming experiments, among them the authors of *Symbolization, Language and Chimpanzees,* generally argue that the implicit assumption of the lexigrams *meaning* something to the apes – a prerequisite for symbolic interpretation - is not something to be overlooked. To make the discussion more insightful, they present a series of four experiments demonstrating how apes may gradually learn the meaning of lexigrams. Initially, the chimpanzees fail to learn anything at all, but after a slight change in setup they manage to produce correct sentences for obtaining a food item or beverage. However, a third experiment shows how these sentences are merely stimulus-action pairs, not symbols. In the final experiment they alter the setting once again in order to induce some of the apes to abandon their straightforward approach and adopt a symbolic learning strategy instead.

Despite the obvious differences, the discussion about ape language has many issues in common with a computer's capability of understanding language. The Chinese Room argument shows how producing a sentence is not necessarily the same as expressing a thought; if this holds for humans and, arguably, for computers, then why not for chimpanzees? If the underlying principles are the same, then we could expect that the learning curves of humans, apes and computer programs are at least somewhat alike. Let us therefore take a closer look at the experiments and results of the chimp language training task, and compare them with a series of experiments conducted with the model presented in chapter four.

Naming Objects

In the first of the four experiments, the chimps are merely required to name shown objects by pressing the correct lexigram token. Contrary to the researchers' expectation – in other studies, chimps were able to perform a similar task (Rumbaugh, 1977) – the apes failed to learn this simple correspondence. Apparently, the purpose of the exercise was not salient enough: a case is reported where a chimp lies down on his back while pressing the right button by mistake, after which he repeatedly tries out this procedure in subsequent trials – presumably under the superstitious impression that not the button caused a reward, but his posture. No matter how obvious the function of a button may be to us, the chimps fail to notice how to operate the lexigram board, likely due to the high amount of possible causes for a reward.

In the second experiment the setup is changed, allowing for the chimps to notice the relation between objects and lexigrams. First, one object is shown while only one button is available. After learning this initial correspondence relation, more buttons and objects are gradually added to the realm of possible lexigram-object combinations. Using this learning method, several apes seemed to catch on to the meaning of the lexigrams and started to construct correct sentences in order to obtain their reward.

The purpose of this chapter is to find a suitable experiment demonstrating the difference between indexical and symbolic interpretation, so let us investigate how to model the chimpanzee learning process with neural networks. Ideally, one would like to present the objects and lexigram board in exactly the same way as the chimps perceive it, for instance by showing the neural network an image of a banana and expecting it to maneuver a robotic arm to press the right button. However, due to practical limitations – visual recognition of objects and robotic manipulation techniques are not in our direct interest here - we will confine the setup to a rather crude simplification of the original experiment, abstracting away from the particular objects used and focusing merely on the characteristics of the different kinds of interpretation processes. To this end, we shall represent both the objects and the lexigrams as binary strings consisting of ten bits, as shown in Table 1.

Network Input	Binary	Correct Output	Binary
Banana$_{\textbf{object}}$	1000000000	Banana$_{\text{lexigram}}$	1000000000
Orange$_{\textbf{object}}$	0100000000	Orange$_{\text{lexigram}}$	0100000000
Apple$_{\textbf{object}}$	0010000000	Apple$_{\text{lexigram}}$	0010000000
Coke$_{\textbf{object}}$	0001000000	Coke$_{\text{lexigram}}$	0001000000
OrangeJuice$_{\textbf{object}}$	0000100000	OrangeJuice$_{\text{lexigram}}$	0000100000
etc.	…	…	…

Table 1 *Binary representation of objects and lexigrams. The network is trained to map corresponding objects to lexigrams, appearing on the same row in this table.*

Note that the binary string representations may be interpreted as icons, in the same way as objects or lexigrams can be icons. Since we are primarily concerned with discovering how the network represents a correlation between two icons, the characteristics of the icons themselves are irrelevant at this point. The object-lexigram index can be modeled using a feed-forward neural network, as explained in the previous chapter. We use a fully connected, three-layer network, with each layer containing ten nodes[10]. The nodes are activated by a weighted sum over the incoming edges, with a step activation threshold function.

Following the training method of the chimp experiment, we start out by using the binary string corresponding to a banana. Once the network has learned to output the correct lexigram – in this case, the string *1000000000* - a second object is added to the set of possible input strings. This continues until all object-lexigram indices have been learned. Note that the network, while learning new indices, also has to remember all previously learned relations.

To train the network, a genetic algorithm is used. A network may be encoded as a string containing all the information about the strengths of its edges[11]. At the start of a learning run,

[10] Considering the input and output strings, using eight nodes per layer would theoretically be sufficient for this experiment as we use a maximum of eight objects. However, because subsequent experiments require networks with up to ten nodes we will use same-sized networks throughout this thesis, allowing for a better comparison of the results.

[11] The phenotypes are encoded as genotypes using Gray encoding, each weight being represented by four bits.

a *population* of such strings is randomly generated. Each individual network is then tested by giving it a number of input sequences and checking what percentage of the output strings is correct. After the entire population of networks has been tested in these trials, the ones with the highest scores *survive*[12] and are passed on to the next *generation*. Finally, new *offspring* is generated in the form of randomly *mutated* variants of their genotype, or by combining two genotypes into one, the biologically inspired *cross-over* process. In each subsequent generation cycle, there is a tendency for the generated offspring to achieve higher scores; eventually, it is likely that a network configuration will be reached in which all indices are represented. The parameters of this genetic algorithm we used for this experiment are displayed in Table 2.

Parameter	**Value**
Input layer neurons:	10
Hidden layer neurons:	10
Output layer neurons:	10
Neuron threshold (θ):	0.40/0.85[13]
Number of children in each generation:	50
Number of elites in each generation:	10
Mutation chance:	1%
Number of learning runs:	100

Table 2 *The parameters of the genetic algorithm, as used for all experiments in this thesis.*

[12] As a selection method, we use *truncated selection*: the highest-scoring individuals are passed on the next generation and allowed to randomly procreate.

[13] The threshold was set to 0.40 when the sum of input values was 1; if the total input values added up to 2, a threshold of 0.85 was used. Ergo, in the naming task and the domain knowledge task, each of the neurons had a lower threshold value. This measure was taken in order to allow for a more fair comparison between similar networks with different kinds of input, and to make the learning algorithm converge faster towards a solution. As the indexical and symbolic tasks have the same threshold value, this difference did not affect the results of those experiments in any way.

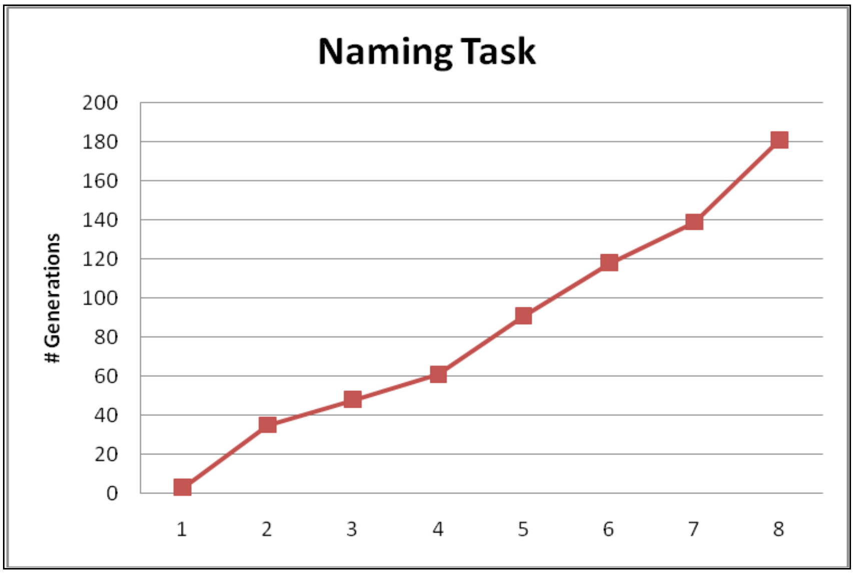


Figure 9 *The results of the naming task experiment, averaged over 100 runs. A neural network is trained to associate objects with lexigrams. For each new object* (x-axis), *the number of generations it took for the network to learn the additional correspondence relation is shown. Note that the learning time will not necessarily increase as more objects are added: an efficient learning algorithm might exploit information redundancies and find a way to learn each object in a fewer number of generations.*

The figure above shows the number of generations it takes for the network to learn a correct mapping from object to lexigram. The results clearly indicate that each additional object significantly increases the time necessary to learn the new correlation. This can be attributed to the limited working memory capacity of the network: the first object to be learned has a relatively high rate of correct network configurations, while the number of possible configurations for learning the fourth index is constrained by the necessity of keeping the former three indices intact.

Learning Symbols

Let's imagine ourselves in a chimp's position for a moment. Suppose we encounter a drink vending machine, which allows us to select a can of soda from a variety of beverages displayed through a glass window. A can may be selected by pressing two digits on a panel attached to the machine. As we have never operated this particular device before, we might be inclined to randomly press several buttons until a can rolls out of the machine. After a number of attempts we will likely notice a pattern: every time some sequence of buttons is pressed, the device dispenses a can of soda. We learn that, in order to obtain a particular drink, we need to memorize the corresponding combination of buttons. However, as the number of learned combinations grows it becomes increasingly difficult to memorize new button combinations. The chimpanzees are faced with a similar problem. It appears that they merely learn an indexical relation between a stimulus object and a lexigram token being pressed. However, the genuine acquisition of a language would require a symbolic bond. Like the apes in other experiments which, it was claimed, could learn over a thousand words, these chimps had learned a set of stimulus-action pairs: the use of tokens

> *... may [...] be simply a set of events which come to precede the receipt of a desired action or object. [...] errorless trials, though given in a fashion which closely approximates that of the final choice, do not lead to symbolic learning even in simple tasks such as food names.*
>
> (Savage-Rumbaugh, 1978, p. 283)

The third experiment conducted by Savage-Rumbaugh was aimed to demonstrate how the sentences constructed by the chimpanzees were actually *holophrases*, that is, sentences that function as words. The ape subjects were required to use the correct verb when requesting a particular food item. For example, when asking for a banana they had to press both the *banana* lexigram and the *give* lexigram. When they wanted orange juice, the *pour* lexigram had to be used. If the apes were using the lexigrams as symbols, one would expect the learning rate to increase when more new objects are added, since the *give* and *pour* lexigrams were already known. However, this was not the case with the experiments: when a new object was introduced, the apes failed to properly use the verbs out of their common context and had to learn the lexigram verb-noun combination from scratch – even though they knew which objects were usually given and which were poured.

We aim to devise a language training experiment for neural networks that shows this kind of behavior. In particular, we want to show that the indexical neural network learns to construct holophrases, while a symbolic network, being able to relate lexigram sentences to its knowledge of objects, will apply a different learning strategy we will call *symbolic learning.*
To train the indexical network, we will alter the learning task of the previous section slightly: instead of words, whole sentences need to be constructed. Therefore, a pattern now represents a concatenation of a noun and a verb. We will once again use a neural network and train it to correlate the binary strings shown in Table 3. Also, a bias unit is added at the end of each input sequence to allow for a valid comparison with the symbolic network. The type of object – e.g. edible or liquid – is alternated after each added object. Figure 10 shows the average number of generations it takes for each object to be learned.

Network Input	Binary	Correct Output	Binary
Banana$_{\textbf{object}}$ + **bias**	1000000001	Banana$_{\text{lexigram}}$ + Give$_{\text{lexigram}}$	1000000010
Orange$_{\textbf{object}}$ + **bias**	0100000001	Orange$_{\text{lexigram}}$ + Give$_{\text{lexigram}}$	0100000010
Apple$_{\textbf{object}}$ + **bias**	0010000001	Apple$_{\text{lexigram}}$ + Give$_{\text{lexigram}}$	0010000010
Coke$_{\textbf{object}}$ + **bias**	0001000001	Coke$_{\text{lexigram}}$ + Pour$_{\text{lexigram}}$	0001000001
OrangeJuice$_{\textbf{object}}$ + **bias**	0000100001	OrangeJuice$_{\text{lexigram}}$ + Pour$_{\text{lexigram}}$	0000100001
etc.	…	…	…

Table 3 *Binary representation of the objects and corresponding lexigram sentences. A network trained on this data will produce the equivalent of a holophrase. Note the slight dip at the third object: because an object of an already known type – an edible, in this case – is added, the network tends to learn the output sentence associated with this object faster. The results are averaged over 100 runs.*

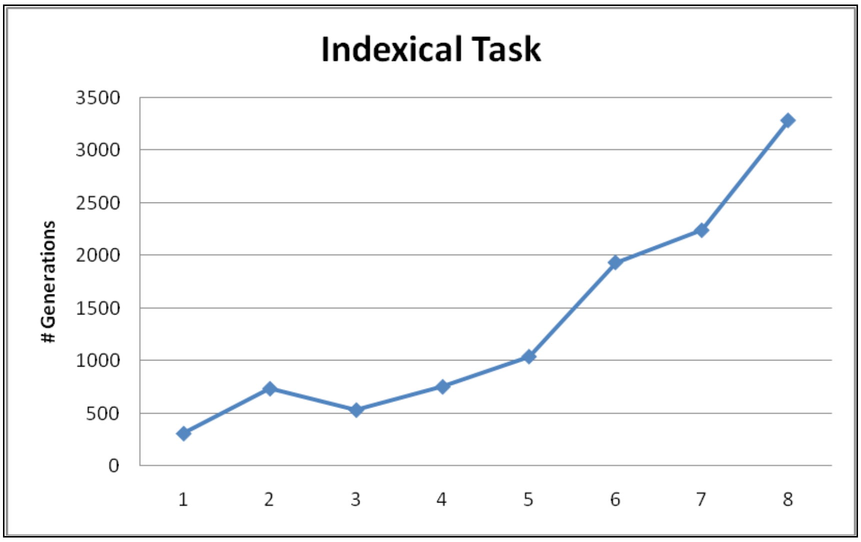


Figure 10 *The learning curve of the indexical sentence construction task, averaged over 100 runs.*

For the symbolic system, a different training set is used; unlike the indexical interpreter, this system's knowledge is not limited to the input object. It also uses its knowledge of the object itself, and how it is related to other objects or actions. We train a network on this domain knowledge using the training data of Table 4. Figure 11 shows the resulting graph.

Network Input	Binary	Correct Output	Binary
Banana$_{\textbf{object}}$	1000000000	Give$_{\text{action}}$	0000000010
Orange$_{\textbf{object}}$	0100000000	Give$_{\text{action}}$	0000000010
Apple$_{\textbf{object}}$	0010000000	Give$_{\text{action}}$	0000000010
Coke$_{\textbf{object}}$	0001000000	Pour$_{\text{action}}$	0000000001
OrangeJuice$_{\textbf{object}}$	0000100000	Pour$_{\text{action}}$	0000000001
etc.	…	…	…

Table 4 *Binary representation of the objects and corresponding actions.*

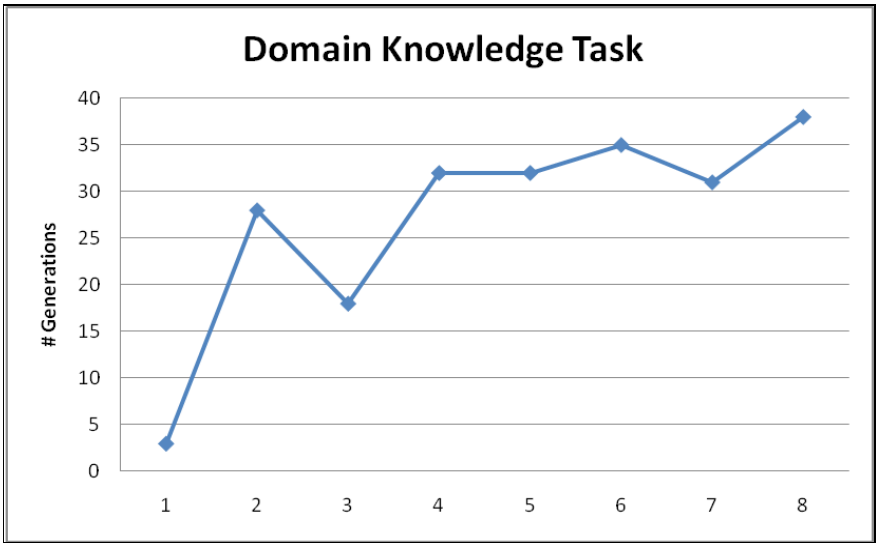


Figure 11 *The results of the domain knowledge task, averaged over 100 runs. Note that it converges to a solution in relatively few generations.*

Now, this domain knowledge may be used by the symbolic network to produce lexigram sentences. If we present the data in such a way to the network that it will notice this higher-order correlation, it may exploit the redundancy in the two indexical systems and use lexigrams to denote objects. Ultimately, we would like to use three neural networks to represent the three indexical systems that consitute a symbolic interpreter. However, in order not to complicate things too much[14], we will assume the two domains have already been learned and incorporated into the training set, and use a single network to learn from the training data shown in Table 5. The data is constructed from concatenated binary strings representing objects, actions or lexigrams, much like the indexical learning task. The only difference is the bias unit being replaced by the output of the domain knowledge task; the symbolic network gains an advantage when it finds the meaning of this extra bit of information by linking the lexigram domain to the object domain by virtue of a higher-order correlation. Figure 12 shows the resulting learning curve of this task.

Network Input	Binary	Correct Output	Binary
Banana$_{\textbf{object}}$ + Give$_{\textbf{action}}$	1000000010	Banana$_{\text{lexigram}}$ + Give$_{\text{lexigram}}$	1000000010
Orange$_{\textbf{object}}$ + Give$_{\textbf{action}}$	0100000010	Orange$_{\text{lexigram}}$ + Give$_{\text{lexigram}}$	0100000010
Apple$_{\textbf{object}}$ + Give$_{\textbf{action}}$	0010000010	Apple$_{\text{lexigram}}$ + Give$_{\text{lexigram}}$	0010000010
Coke$_{\textbf{object}}$ + Pour$_{\textbf{action}}$	0001000001	Coke$_{\text{lexigram}}$ + Pour$_{\text{lexigram}}$	0001000001
OrangeJuice$_{\textbf{object}}$ + Pour$_{\textbf{action}}$	0000100001	OrangeJuice$_{\text{lexigram}}$ + Pour$_{\text{lexigram}}$	0000100001
etc.	…	…	…

Table 5 *Binary representation of the input and output sentences of the symbolic task. Although there is only minimal difference with the indexical task – the input string is sometimes altered by one bit - this extra domain knowledge is enough to cause a significantly more efficient learning strategy.*

[14] To limit the scope of this thesis, a setup with three indexical networks will not be treated here. However, the potentials of such a model will be shortly discussed in the final part of this thesis.

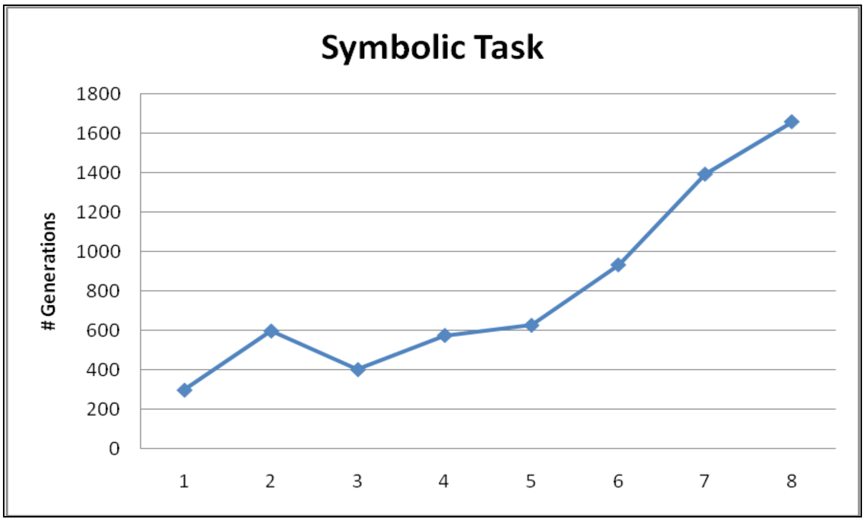


Figure 12 *The learning curve of the symbolic task, averaged over 100 runs.*

Comparison of Results

Let’s reconsider our example with the soda vending machine. We have seen how memorizing key combinations taxes working memory, since every single holophrase has to be remembered. However, if there is an underlying logic to the key combinations, we can employ a different learning strategy which allows us to make use of the glass window displaying the cans. Suppose the cans of soda behind the glass are all lined up in an orderly fashion, with the top row containing only coke brands, the row below it being filled with different orange juice cans, and so on. Suppose the two buttons that need to be pressed indicate the respective row and column of the can that will be dispensed. If we carefully study the relation between the process inside the machine and the button combinations, we may suddenly discover the symbolic meaning of the buttons: the first number is a symbol for a type of drink. No longer will we string the buttons together into a holophrase, but each of them is used as a separate word in a sentence.

The fourth and final chimp experiment follows a comparable strategy. The apes’ attention is drawn towards the food and drink dispensers by increasing their saliency both audibly and visually. The chimps now notice a dispenser opening, even if it is empty. This allows them to draw conclusions about the syntactic and semantic soundness of the produced sentence. Two out of four chimpanzees switched to a different learning strategy: instead of memorizing all possible holophrases, they started paying attention to the relation between the two – object and lexigram - domains. It is argued that this difference can be ascribed to the learning strategy of the apes: some of them, producing a kind of stimulus-response holophrase, had learned the task through indexical learning. Others used symbolic learning to relate their object knowledge to the task.

Sorting Objects:

Chimp Name	Total Trials to Training Criterion	Total Errors During Training	Test (correct/trials)
Lana	160	19	10/10
Sherman	1115	200	not given
Austin	1210	252	not given

Labeling Objects:

Chimp Name	Total Trials to Training Criterion	Total Errors During Training	Test (correct/trials)
Lana	1493	199	3/10 (retest: 1/10)
Sherman	852	68	9/10
Austin	3239	429	10/10

Table 6 *Chimpanzee learning curves on two different tasks: sorting objects vs. labeling objects (Savage-Rumbaugh, 1978). The sorting task merely required iconic skills, while the object labeling task was setup in such a way that a symbolic learning strategy was advantageous.*

Table 6 above, taken from the original chimp training article, contains the training results of a similar experiment for three chimpanzees. Although the amount of data is rather limited, the results clearly show a difference between *Lana* on the one hand and *Sherman* and *Austin*, who have allegedly adopted a symbolic learning strategy, on the other. Lana can easily sort different objects without any error, a task that is far more difficult for Sherman and Austin. On a superficial level, Lana appears to have grasped the training data at a faster rate than the other two chimps; however, on a task requiring a deeper understanding of the objects involved, she performs quite differently. Even after extensive training she consequently fails to label objects correctly. The researchers conclude that Sherman and Austin, contrary to Lana, take their time to understand the underlying system, resulting in much higher test scores.

We can discern the same patterns from the neural network experiments described in the previous sections. Our indexical interpreter makes the same kind of superficial connection between an input object and a correct output sentence - a holophrase. The symbolic network uses its domain knowledge to understand the system underlying the token manipulation, in order to discover the meaning of these symbols. Although we might presume this domain knowledge to be readily available to the network, we will add the training time for this network to the symbolic network training time, since we aim to make a fair comparison. The resulting graph, containing the learning times of all experiments, is presented below.

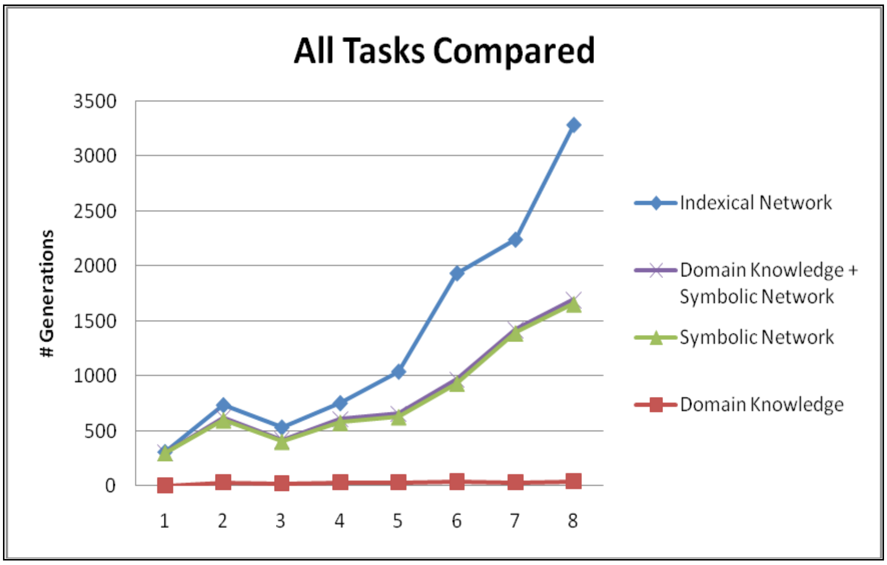


Figure 13 *The results of the indexical and symbolic networks compared, averaged over 100 runs. Note that the domain knowledge task takes an almost trivial amount of time to learn.*

When comparing these graphs to the data from the tasks in Table 6, one may notice a similarity and a difference. In both cases, the symbolic approach turns out to produce the most efficient results in the long run. Even if the training time of the domain knowledge task is added to the symbolic network's learning curve, the amount of time required to learn a new sentence is roughly half the time an indexical interpreter needs for the same task. The chimp experiments show a similar tendency. But the results differ in another respect. While the symbolic chimps need time to discover the relation between the objects and the lexigrams, the symbolic network outperforms the indexical network straight from the start. This dissimilarity can be attributed to the simplification we mentioned earlier: we have used a single feed-forward network instead of three distinct networks. We have made sure that the training data has been setup in such a way that the network will quickly discover the higher-order link. In many situations, such as the chimp experiment, this kind of neatly structured data will not be directly available and the network will have to search for new correlations among its indices, thereby slowing down the learning process.

Naturally, the differences between chimp brains and neural networks are plentiful – not only do they differ in size, but also in features such as network architecture or type of input - making it hard to compare the two. However, the purpose of these experiments is not simply to show their likeness. Rather, we intend to unveil an underlying logic that applies to all these experiments, a logic in which we can express the general differences between indexical learning and symbolic learning. This logic, captured by Peirce’s semiotics, allows us to investigate the structure of neural networks, experiment with their behavior and compare the resulting learning curves. The future research potential of using this underlying logic will be briefly discussed in the final chapter of this thesis.

Chapter 6: Stepping down the Merry-go-round

Conclusion

> *"No one suspects that when a parrot says, "Pretty bird!" he really "knows" that these sounds are supposed to refer to his (or another bird's) appearance. He has learned to mimic the sound of the words, that's all. Sometime in the past he was rewarded for producing this phrase [...] and he now produces it spontaneously. But what if he is taught to say, "Wanna cracker!" and rewarded with a cracker every time he says it? Presumably, when he wants a cracker, he will say so. Is this different? Should we now say that he knows what the words mean?*
>
> (Deacon, 1997, p. 25)

The use of language is closely related to *meaning* and *understanding*. Words and other symbols cannot be seen apart from the interpreting systems that attach meaning to these tokens. The parrot example above, taken from *The Symbolic Species* (Deacon, 1997), shows how a sensible use of tokens does not automatically imply a correct understanding of such a sentence. Although the parrot's request may appear quite reasonable to us, it has not attached any meaning to the words – the parrot's utterance is merely a pre-learned action invoking the distribution of a cracker. We have argued throughout this thesis that the difference between such a straightforward response and a genuine understanding of symbols is an important distinction, one that is regularly overlooked.

The Chinese Room Argument suggests that the same difference also applies to human and computer symbol processing. Some disagree with Searle's argument, claiming that it is too unspecific or founded on a misconception of human brain processing; however, we have shown that at the root of the argument lies a misconception of the word *symbol*. Computers manipulate tokens following a fixed set of indexical rules. However, as they lack the symbolic skill of finding new relations among networks of indices, one cannot maintain that computers are symbolic interpreters; rather, they treat tokens as icons or indices. This claim is supported by the Symbol Grounding Problem: it is theoretically impossible for computers to understand symbols, as a translation of meaningless tokens into other meaningless tokens merely leads to infinite regress. The main research question of finding a symbolic representation now becomes even more relevant to AI: if humans can use symbols but computers can't, then what kind of artificial system *is* capable of symbolic interpretation?

We revert to the theory underlying interpretation. The interpretant, being the mental reaction of an interpretation process, is itself a potential new sign for another interpretation. In chapter three we have learned how the three kinds of signs – *icon*, a sign based on similarity; *index*, based on association and *symbol*, based on convention – are hierarchically related. Each and every interpretation starts out with a recognition process, an iconic interpretation. The iconic interpretant forms the starting point for an indexical interpretation, which ends with another icon. The two are indexically linked by virtue of a higher-order icon: the similarity of all the occasions in which they occurred together. Thus, an index consists of a triadic relation of icons; much in the same way, a symbol consists of a triadic relation of indices. Two indices represent the associative relations in their respective domains. A third, higher-order index links these two together, based on a particular iconism among both networks of indices.

Due to the hierarchical nature of signs, it is required for any symbolic system to possess all three kinds of interpretation skills. In order to design an artificial symbolic system and at the same time show how its behavior contrasts with an indexical system, we choose to model the three interpreters using neural networks, allowing for a valid comparison among these models. The typical iconic neural network consists of a feed-forward network architecture, able to classify incoming data. Furthermore, it is argued that a recurrent neural network is a plausible candidate for indexical interpretation, although, given the right training data, a feed-forward structure may also be used. Considering the scope of the main research question, we opt for the less complicated, second option. We will, however, briefly touch upon the subject of recurrent networks in the following section.

Finally, in order to model a system capable of representing symbols we have formulated a new type of network architecture. Two different indexical networks represent two domains of indices, while a third relates them by virtue of their commonalities, allowing for a mapping from one domain to the other. As such, we find this hierarchical structure, which we will call Emerging Neural Network, to be a fitting structure for symbolic representation: the higher-order network gains *insight* into the topology of the indexical relations of the networks below. Using the redundancy of this information it links the two domains together, thereby creating a framework of symbolic reference with a degree of freedom unprecedented by the indexical system. Again, we aim to keep the experiments uncomplicated and model the Emerging Neural Network using a feed-forward architecture.

The results[15] of the language training task in the final chapter demonstrate a significant difference between the indexical and symbolic neural network. The former shows a steep learning curve, as learning time increases each time new sentences are added to its vocabulary. While the latter learns the first few sentences at roughly the same rate, it soon notices a correlation among the indices of the object and lexigram domains. Using the redundancy of this data, the symbolic system promptly becomes able to learn new sentences at a much faster rate than its indexical counterpart.

[15] See Figure 13

Discussion

Considering the extensiveness of the topic and the broad range of disciplines required for this study, we will not only provide a summary of the results but also an evaluation in which the outcome is matched against our original goals. So, to what extent *have* we found an answer to the main research question? Is it fair to say that we have developed a model for representing symbols, and shown its advantages over non-symbolic systems? As we had predicted in the introduction, the answers to these complex questions are not straightforward.

The definition of the Symbol Grounding Problem suggests that the cause of a computer's incapability for understanding language lies in the absence of a link between its perceptions and its symbol representation. It lacks a sub-symbolic, intermediate layer that connects both domains. Harnad argues that this layer could never be established by redefining existing symbols: since those rewrite rules will only lead to other symbols, the original symbols will never be *grounded* in perceptions – the so-called *symbolic merry-go-round*. Clearly, a new approach is required in dealing with this problem. In order to find an effective model for the intermediate layer, we have studied the symbol interpretation capabilities of the computer. Our semiotic analysis has shown that computational symbols are, in fact, icons. Although computers deal with tokens that have meaning to *us*, this does not necessarily imply a relation between the tokens and their referents exists *for the computer*. It merely treats these tokens as icons and indices. Given the vertical structure of the sign hierarchy, the computer's symbolic qualities need to be built on a foundation of icons and indices, allowing for the symbols to emerge from this sub-symbolic layer. Therefore, we have proposed to adopt a bottom-up approach to model this intermediate layer from icons to symbols.

Using neural networks, we have shown how to translate this hierarchically layered model in a concrete structure. This newly developed layered neural network demonstrates how *insight* in its own interpretation process is an essential element of symbolic interpretation, and opens up the possibility of designing a computer model with symbolic qualities. Note that creating such a computer simulation leaves open the question whether computers *themselves* can represent symbols. But we will not treat this question here, as it opens up a discussion about the ontological nature of simulations that lies far beyond the scope of this research. The point we want to make is that models such as neural networks are fit for representing symbols – possibly, even more fit than regular computational structures.

The goal of the experiments was to test whether a significant difference exists between the interpretation qualities of the indexical and symbolic systems, as part of the main research question. However, the acquired results should not be regarded as definite proof for finding a structure that adequately models the sign hierarchy. Rather, it serves as a proof-of-concept, indicating that the hierarchical composition of symbols is a universal trait of symbol systems, both in neural networks and chimpanzees. After all, the underlying semiotic theory and the correct transformation to a neural network model are examined in the experiments. It is important to note that several simplifications to the model have been made in these experiments: we have imposed boundaries on this model in order to reduce its complexity, by explicitly representing the domain knowledge in the learning data table. Although this setup allowed us to use a simple feed-forward network, facilitating an easy comparison among the different models, it is not yet as sophisticated as we want it to be. Eventually, the symbolic neural network needs to be able to create new conventions without any preconceived learning data – ideally, it recognizes these relationships amongst its own interpretation processes without any preconceived learning *goals*. However, in order to achieve this, the theory describing these emerging neural networks requires additional research.

We can attempt to point out the general directions towards which continued research might develop. Concerning the iconic interpreter, we have already seen how classification is a common task for many types of neural networks. The option of using recurrent neural networks to learn indexical connections is hinted at in chapter four, as (Elman, 1990) shows their aptitude for learning associations. Even though our experiments show a feed-forward network can represent indices as well, the recurrent system appears to be able to handle less structured data better. Similarly, the symbolic network should theoretically be composed of three hierarchically ordered indexical networks to allow for a genuinely symbolic interpretation of its perceptions. How exactly such a meta-network will operate on the other two networks, is an open question that remains to be answered. However, we can suggest that the realization of such a model will likely be based on cybernetics, self-learning systems research and emergence theory.

The Emerging Neural Network theory presented in this thesis has the potential for filling the gap between symbols and perceptions that is exemplified by the Symbol Grounding Problem. It provides a basic prototype for a sub-symbolic system, consisting of icons and indices, out of which symbols can emerge. It explains why a symbolic system demonstrates a different learning behavior from a purely indexical system's behavior. We have presented a model for which stepping down the symbolic merry-go-round is not a problem - it simply never gets on the ride in the first place.

Acknowledgements

First and foremost, I would like to thank Terrence Deacon for the wonderful time I have been able to spend at Berkeley. It was an amazingly educative experience, and I hope to continue learning from our discussions in the future. Fellow “Pirates” Julie Hui, Ty Cashman, Jeremy Sherman, James Haag and Jay Ogilvy: thank you for the many interesting social gatherings.

Furthermore, I want to extend my gratitude to John Searle for allowing me to attend his discussion sessions and to Liane Gabora, who helped a great deal in getting this research on track.

Back on the other side of the Atlantic, thanks to Marco Wiering and John-Jules Meyer for supervising this project. I am also grateful to the many other colleagues, friends and family whom I have been able to exchange thoughts with on this interesting subject: Harm Aarts, Walter de Back, Renate Bartsch, Paul Brinkkemper, Tim van Erven, Annabel Goth, Jasper ’t Hoen, Merlijn Hurx, Geert Jonker, Andrew Koster, Wouter LeFêvre, Natalie Leijnen, Puck Leijnen, Sieuwert van Otterloo, Rina Smulders, Sven Stringer, Paul Vogt, Jilles Vreeken, Jelmer de Vries, Silvain van Weers - Thank you all.

List of References

Bateson, G. (1979) *“Mind and Nature”*, E. P. Dutton, New York.

Chandler, D. (2002) “*Semiotics: The Basics”,* Routledge.

Deacon, T.W. (1997) *“The Symbolic Species: the co-evolution of language and the brain”*, W. W. Norton & Co., New York.

—— (2003) *“Universal grammar and semiotic constraints”*, In Christiansen, M. & Kirby, S., ed., *Language Evolution*, 7, pp. 111-139. Oxford University Press.

Dreyfus, H.L. (1981) *“From Micro-Worlds to Knowledge Representation: AI at an Impasse”*, In Haugeland, J., ed., *Mind Design: Philosophy, Psychology, Artificial Intelligence*, pp. 161–204. MIT Press, Cambridge, MA.

Elman, J.L. (1990) *“Finding Structure in Time”*, Cognitive Science, 14-2, pp. 179-211.

Harnad, S. (1990) *"The Symbol Grounding Problem"*, Physica, D 42, pp. 335-346. Eprint.

Holland, J.H. (1975) *“Adaptation in Natural and Artificial Systems”*, University of Michigan Press, Ann Arbor.

Hookway, C. (1985) *“Peirce”*, London, Routledge & Kegan Paul.

Hui, J., Cashman, T. & Deacon, T.W. (2006) *“Bateson's Method: Double Description. What is it? How does it work? What do we learn?”.*

Kohonen, T. (1982) *“Self-organized formation of topologically correct feature maps”*, Biological Cybernetics, 43, pp. 59-69.

McCarthy, J. & Hayes, P.J. (1969) *“Some philosophical problems from the standpoint of artificial intelligence”,* Machine Intelligence, 4, pp. 463-502.

McCulloch, W. & Pitts, W. (1943) *“A logical calculus of the ideas immanent in nervous activity”*, Bulletin of Mathematical Biophysics, 7, pp. 115-133.

McDermott, D. (1981) *“Artificial Intelligence Meets Natural Stupidity”,* Mind Design: Philosophy, Psychology, Artificial Intelligence, pp. 143-160. MIT Press, Cambridge, MA.

Newell, A. & Simon, H.A. (1976) *"Computer Science as Empirical Inquiry: Symbols and Search"*, Communications of the ACM archive, 19-3, pp. 113-126. ACM Press New York, NY, USA.

Ogden, C.K. & Richards, I.A. (1923) *"The Meaning of Meaning"*, 8th Ed. New York, Harcourt, Brace & World, Inc.

Panofsky, E. (1972) *"Studies in Iconology: Humanistic Themes in the Art of the Renaissance"*, Harper & Row, New York.

Peirce, C.S. (1894) *"What Is a Sign?",* In The Essential Peirce: Selected Philosophical Writings, vol. 2, p. 9.

—— (1955) (Justus Buchler, ed.) *"Philosophical Writings of Peirce"*, Ch 7. Dover Publications, New York.

Rumbaugh, D. (1977) *"Language Learning by a Chimpanzee: The Lana Project"*, New York, Academic Press.

Rumelhart, D.E. et al. (1986) *"Learning Internal Representations by Error Propagation"*, In Parallel Distributed Processing, vol. 1, pp. 318-362. MIT Press, Cambridge, MA.

Savage-Rumbaugh S. & Rumbaugh D.M. (1978) *"Symbolization, Language and Chimpanzees: A theoretical reevaluation on initial language acquisition processes in four Young Pan Troglodytes"*, Brain and Language, 6, pp. 265-300.

Searle, J.R. (1980) *"Minds, brains and programs"*, Behavioral and Brain Sciences, 3-3, pp. 417-457.

Turing, A. (1950) *"Computing machinery and intelligence"*, Mind, vol. LIX, no. 236, pp. 433-460.

Vogt, P. (2002) *"The Physical Symbol Grounding Problem"*, Cognitive Systems Research, vol. 3, 3, pp. 429-457.

Winograd, T. (1984) *"Computer Software for Working with Language"*, Scientific American, vol. 251, no. 3, pp. 90-101.