to be published in: Proceedings of Trends in Special Language and Language Technology, R. Temmerman (ed.) (Standaard Publishers, Brussels, 2001)

Mining Associative Meanings from the Web: from word disambiguation to the global brain

Francis HEYLIGHEN

CLEA, Free University of Brussels, Pleinlaan 2, B-1050 Brussels, Belgium

E-mail: fheyligh@vub.ac.be, http://pespmc1.vub.ac.be/HEYL.html

ABSTRACT. A general problem in all systems to process language (parsing, translating, etc.) is ambiguity: words have many, fuzzily defined meanings, and meanings shift with the context. This may be tackled by quantifying the connotative or associative meaning, which can be represented as a matrix of mutual association strengths. With many thousands of words, there are billions of possible associations, though, and there is no obvious method to measure all of them. This "knowledge acquisition bottleneck" can be tackled by mining implicit associations from the billions of documents and millions of users on the World-Wide Web. The present paper discusses two methods to achieve this: lexical co-occurrence, a measurement of the frequency with which words appear in each other's neighborhood, and web learning algorithms, an application of the Hebbian rule to create associations between subsequently "activated" words or pages. The mechanism of spreading activation can be applied to the resulting associative networks for clustering, context-driven disambiguation, and personalized recommendation. A generalization of such methods could transform the web into a "global brain", that is, an intelligent, learning network that assimilates the implicit knowledge and preferences of its users.

1. Introduction

In the present invited paper, I wish to look at future, rather than existing, language technologies. I will review some fundamental problems that confront all computer systems for processing language, and, with reference to some recent studies, demonstrate methods to tackle them. While these approaches appear capable of solving the problem in principle, it will, however, still require a lot of work to transform these general principles into reliable applications. Therefore, the focus of this paper will be conceptual rather than practical, attempting to explain the thinking behind some technological trends that are likely to become important in the next few years.

Perhaps the most fundamental problem in language processing is *ambiguity.* Words, phrases and sentences can have many different meanings, and these meanings shift with the context [Heylighen & Dewaele in press]. People have usually little difficulty in grasping the right meaning. Computer programs, on the other hand, are easily confused, and tend to interpret a language fragment in one way, when it should have been interpreted in another way. This problem of disambiguation recurs in the most diverse domains [Ide & Véronis 1998]: parsing (the same word can have different grammatical roles depending on its meaning), speech recognition (similar sounding words may actually be different), information retrieval (search words may refer to very different subjects), natural language comprehension (a command given to a computer can be misinterpreted), and content analysis.

The ambiguity problem was perhaps recognized most early in machine translation [Bar-Hillel 1960]. In spite of the great advances in computer technology and in Artificial Intelligence (AI), reliable translations still require human involvement. Yet, in the 1950's, when the first AI programs were developed, it all seemed simple: for every word in the source language you need to find the equivalent in the target language, using a correspondence list similar to a translating dictionary. The only thing more you would need is a program that understands enough of syntax to put the words in the right order. Anybody who has done any translation work knows that things are not that easy. The same word can mean many different things, depending on the phrase, the sentence, or even the document in which it appears, and each meaning requires a different translation.

Further developments in AI have learned that lesson, gradually taking into account more and more about the context and even the complete world-view or model that a languageuser keeps in mind when interpreting a sentence. For example, consider the sentence: “He saw the seagulls sitting on the bank”. Although the most common meaning for *bank* is "financial institution", it is obvious from the sentence that here we should interpret it in the much less common sense of "edge of a river". An AI program that knows about seagulls should be able to deduce that these birds rarely sit on financial institutions, but frequently do sit on river edges. But this is a specialized knowledge about concrete aspects of the world, which cannot be found in grammars or dictionaries. Thus, for a computer program to correctly interpret a phrase, you need more than purely lexical or grammatical knowledge, you need to have an understanding of the world in general, and of the subject of the language excerpt in particular. For that reason, the problem of word disambiguation has been called "AI-complete", i.e. solving it would require solving all outstanding deep problems of artificial intelligence [Ide & Véronis 1998].

The difficulty is that even the basic knowledge we all use in our everyday actions is huge, complex, and difficult to formulate explicitly. The most ambitious attempt to date to codify this common sense knowledge in a computer system is the CYC project, a team of hundreds of people led by AI pioneer Douglas Lenat [Lenat & Guha 1990]. Yet, in spite of thousands of man-years of effort, the CYC project still has not achieved its aims. The core of the problem is what AI researchers call the “*knowledge acquisition bottleneck*”. Present-day computers can easily manipulate millions of stored data items. The human brain too stores and processes millions of data without effort. The difficulty, however, is to transfer this knowledge from a brain into a computer.

This transfer is hampered by the completely different way in which brains and computers store data [Heylighen 1991]. Humans develop knowledge by repeatedly experiencing phenomena together, thus learning associations between the concepts that represent these co-occurring phenomena. Computers must receive their knowledge in the form of explicit, formal statements, consisting of discrete symbols. Human learning goes on continuously since birth, most of the time without noticeable effort. For a person to formulate a piece of knowledge in the form of a proposition that can be understood by a computer, on the other hand, requires quite some effort, since fuzzy, implicit impressions must be converted to unambiguous, logical truths. Moreover, most of our intuitive, associative knowledge cannot even be expressed in the form of logical propositions. For example, how would you produce a proposition expressing the fact that, in your experience, seagulls are more at home around rivers than around financial institutions? Categorical rules such as “if a bird is a seagull, then it is situated near a river” or “if a bird is situated near a financial institution, then it is not a seagull” are obviously in many cases incorrect.

An apparent alternative might be to formulate a statistical rule, expressing the probability that you would find a seagull in a given situation. But the problem then is how to determine the probability value: it seems rather pointless to collect statistics about the number of seagulls sitting on financial institutions or on river edges. Perhaps you could still gather a reasonable statistic if you only needed to know the relation between seagulls and rivers. But what about the billions of other possible associations that you would encounter when trying to interpret any natural language fragment? What about the probability of finding a penguin near to ice? Or how likely is it to find a grandmother in a financial institution? Associations are not probabilities: they are much more fluid, subjective, and difficult to grasp.

Does this mean that computers will forever remain far behind humans in their capacity for understanding natural language? The message I wish to convey in this paper is more optimistic: it is possible to represent intuitive, associative knowledge in a computer, and the knowledge acquisition bottleneck can be overcome. First, I will show how associative meanings can be modelled mathematically. Then, I will discuss different techniques through which the enormous reservoir of information available through the World-Wide Web can be “mined” to extract such associations [cf. Bollen, Van de Sompel & Rocha 1999]. I will finally suggest some possible approaches to use the thus acquired knowledge in order to tackle problems such as word disambiguation.

2. Representing associative meaning

According to the most traditional view of language or representation, the meaning of a word corresponds to the thing or category of things that the word stands for. For example, the word “seagull” stands for, or denotes, a particular group of birds. This may be called the denotative meaning or *denotation* of the word. In logic, this type of meaning is usually called the *extension* of a symbol. One difficulty with this approach is that the underlying theory of knowledge, where concepts or symbols are supposed to reflect outside entities, runs into a host of philosophical problems [Heylighen in press]. A more practical difficulty is that language interpreters, whether humans or machines, in general do not have access to the external objects that the words are supposed to denote. Therefore, the denotative view of meaning is of little help in determining what a word refers to.

When working with texts, we generally only have access to words, not to things. Therefore, it would be better if we could determine the meaning of a word only by using its relations to other words. Let us call this second type of meaning *connotation*. According to the [American Heritage Dictionary 1996], connotation can be defined as:

> *An idea or meaning suggested by or associated with a word or thing. The set of associative implications constituting the general sense of a word in addition to its literal sense. Logic: the set of attributes constituting the meaning of a term; intension.*

The problem now is to determine connotation in such a way that it could be embedded in a computer program, allowing that program to correctly interpret a word in a given context, taking into account all the different meanings or associations that same word can have. Let us begin by defining word connotation more precisely as: *the whole of associations a word has with other words*.

An association can then be expressed as a relation between any two words or concepts with a strength varying between, e.g., 0 and 1. An association of 1 between two words would mean that, whenever we encounter the one word, we are immediately and unequivocally reminded of the other word. This seems like an extreme case, that, perhaps may only be encountered with two of the purest synonyms. A more common association of 0, on the other hand, would mean that given the one word, we would not in any way be predisposed to think of the other word, although the words need not be mutually contradictory or incompatible. Association 0 would simply mean that two words are independent, like *seagull* and *financial institution.* Encountering the word *seagull* does not exclude us from thinking about *financial institution*. It is just that, if you want to make somebody think about a *financialinstitution*, the clue *seagull* would be of no help whatsoever. *Seagull* and *river*, on the other hand, do have a positive association, although it may not be particularly strong. *Cat* and *mouse*, or *mouse* and *cheese*, are examples of stronger associations. The (transitive) association between *cat* and *cheese* would be quite weak, on the other hand.

Associations are in general not reversible or symmetric. *Mouse* may remind you strongly of *cheese*, but *cheese* has a much weaker association with *mouse*. Thus, the association *mouse* → *cheese* might have strength 0.5, whereas *cheese* → *mouse* might only score 0.2. Note that the absolute value of an association is not so important, but only its value relative to other associations. Thus, a score of 50 % for *mouse* → *cheese* will not mean much on its own, but the fact that this score is significantly larger than the 10% score for *mouse* → *meat* is significant. It means that if you get a sentence of the form *The mouse ate the X*, where *X* is an ambiguous word that you have difficulty interpreting, then, out of two possible interpretations, *cheese* and *meat*, you will be strongly inclined to choose the former.

The list of all possible associations between words in a language can be represented as a matrix $A = (a_{ij})$, where a_{ij} is the strength of association going from word w_i to word w_j. The connotation or associative meaning of a word w_i can then be represented by the list of all associations w_i has with other words: $(a_{i1}, a_{i2}, a_{i3}, ... a_{in})$. This is the i-th row vector of the association matrix A. Thus, every word can be represented by a list, or vector, with n components, where n is the total number of words in the language or corpus. This vector situates the word as a point in an n-dimensional semantic space of possible meanings [cf. Burgess 1998].

Note that the meaning of a word is thus expressed by the whole of its associative relations with other words. These words too have their meaning defined by their relations with further words, including the word we started with. Thus, meaning is determined in a *bootstrapping* way: there are no semantic primitives, there is no independent “ground” or “foundation” by which meaning is supported; instead, meanings mutually determine each other. Although this may seem circular, bootstrapping methodologies allow us to recover various basic structures in knowledge and language [Heylighen in press; Finch & Chater 1992; Cairns et al. 1997], as we will illustrate when we discuss clustering.

3. Measuring associations

If associations can be expressed by numbers, the next question is how to measure them. One obvious method is by submitting a questionnaire to a group of subjects. The questionnaire lists different possible associations, such as *mouse* → *cheese*, *mouse* → *cat*, *cat* → *cheese*, etc., each followed by a list of possible association strengths, e.g. 0% - 20% - 40% - 60% - 80% - 100%. You then ask the subjects to circle the number that corresponds best to what they intuitively perceive as the degree of association. Instead of numbers, you could also use qualitative estimates such as “not at all” - “a little” - “moderately” - “strongly” - “completely”, or a graphical ruler or line on which subjects could indicate their estimate of strength. The exact manner does not matter that much, since there would anyway be much individual variation between the subjects, and the only significant result would be the average score over a sufficiently large group of subjects.

Although this method may work well if you need only a few associations, it is obviously impractical if you need to establish all connotative meanings in a realistic language sample. Suppose that your lexicon contains 10,000 words, then you would need to determine 100,000,000 association strengths. A computer may have little difficulty storing and processing such large numbers, but no human volunteer would be willing to score millions of cross-associations. Here, we are again confronted with the knowledge acquisition bottleneck. The problem becomes somewhat less daunting if we note that most associations will have strength 0, because the corresponding words are basically independent: the association matrix is "sparse". Thus, instead of comparing all possible couples of words, we could limit ourselves to those cases where there is a straightforward association.

3.1. Word association norms

A well-known method to do this is the psychological technique of free association. The observer proposes a word to the subject, who responds with the first word that comes up in his or her mind. For example, when the observer says “mouse”, the subject answers “cheese”, “Mickey”, or “cat”. The same words are presented to a large group of subjects, and the responses are collected for each word. From the list of responses, all the words that were proposed by only one person are eliminated, as these are likely to be idiosyncratic associations that have meaning only for this particular subject. The words that appear more than a minimum number of times are counted, and receive an association strength proportional to the number of times they were uttered. This method has been used to determine so-called “word association norms” for the most common English words [Palermo & Jenkins 1964; Moss & Older 1996; Nelson, McEvoy & Schreiber 1998]. Some example results can be found in Table 1.

cue word	**association**	**association strength**
seagull	*bird*	0.480
	beach	0.209
	ocean	0.041
	shit	0.020
	fish	0.014
	fly	0.014
	pest	0.014
	sea	0.014
	water	0.014
	white	0.014
confess	*tell*	0.232
	admit	0.119
	church	0.086
	sin	0.053
	deny	0.046
	lie	0.046
	truth	0.046

Table 1: word association norms for the words *seagull* (complete) and *confess* (top half), excerpted from [Nelson, McEvoy & Schreiber 1998]. The association strength represents the proportion (e.g. 48 %) for a particular association (e.g. *bird*) on the total of all associations produced (e.g. *bird*, *beach*, ..., *white*) in response to the cue word (*seagull*)

One problem with this methodology is that since users are prodded to react quickly, without thinking about degrees of relatedness between words, they often produce words that are associated only superficially, e.g. because they rhyme, sound similar, or are part of a common collocation (e.g. *Mickey Mouse*). A more serious problem is that this method only recovers the dozen or so most salient associations for any given word, while ignoring the weaker, but still significant ones. For example, there exists a weak association between *mouse* and *dog* as these are both mammals that live with people, and thus are more likely to be encountered in the same context than two randomly chosen words, such as *mouse* and *bank*. Yet, it is quite unlikely that someone would respond *dog* when prodded with *mouse*.

The problem remains that directly eliciting all possibly relevant associations from human subjects requires an inordinate amount of effort. This problem could be evaded if we could somehow automate the process, and let the computer measure associations. Yet, associations originate in the human mind, to which the computer does not have direct access. Still, associations do not remain hidden in the depths of our brain: they show up in the way we use language. What remains to be done then, is to find a sufficiently rich source of data on language use, and to develop an efficient method to extract associations from those data. Thanks to the explosive development of network technologies, the data source is at hand: the World-Wide Web. The web presently contains billions of natural language documents. These documents are being used every day by hundreds of millions of people, who apply their implicit knowledge of the language to interpret the meaning of these documents, and to decide about their further actions in the web. I will now discuss two methods to mine this abundant resource, one using the static structures of texts, another one using the dynamic choices made by people.

3.2. Lexical co-occurrence

A simple way to estimate the degree of association consists in counting how often two words appear together, in the same document or in the same phrase. The assumption is that if language users often hear or read two words together, then these words will become associated in their minds, and they will in turn tend to use these words together in their speaking and writing. For example a text about *mice* is more likely to contain the word *cheese* than the word *meat.* [Spence & Owens 1990] and [Wettler & Rapp 1993] have confirmed that such lexical co-occurrence correlates with association strength, as derived from word association norms.

Just counting the frequency with which words appear is not sufficient, though, as some words are very frequent (e.g. the word *the*) and will appear in a very large number of language fragments, while others (e.g. *Antarctica*) are quite rare, but will still be strongly associated with other rare words (e.g. *penguin*), and therefore frequently occur in the same documents. One solution is to calculate the association strength from word w_i to word w_j as the *conditionalprobability* that you would find word w_j, given a text that contains word w_i :

$$a_{ij} = P(w_j | w_i) = \frac{P(w_i \& w_j)}{P(w_i)} = \frac{N(w_i \& w_j)}{N(w_i)}$$

$P(w_j)$ stands here for the probability that a text would contain word w_j , $P(w_i \& w_j)$ for the probability that it would contain both w_i and w_j , $N(w_i)$ for the total number of texts in the sample that contain w_i, and $N(w_i \& w_j)$ for the total number of texts that contain both w_i and w_j . A related, commonly used formula is (pointwise) *mutual information* [Church & Hanks 1990], but this has the limitation that the results are necessarily symmetric in *i* and *j*:

$$a'_{ij} = \log \frac{P(w_i \& w_j)}{P(w_i).P(w_j)}$$

To calculate such association strength between any pair of words, it is sufficient to count how many documents contain any single word in the list, and how many documents contain any pair of words. This is automatically done by the kind of “web robots” used to build search engines, that download large numbers of web pages, and index their lexical content. In fact, when you enter a particular word or conjunction of words in a search engine, you usually get an estimate of the number of documents that contain either the word or the conjunction, and that is all the data you need to build an extensive association matrix. To test this approach, I entered some word combinations in the Google search engine, and using the conditional probability formula on the number of results I calculated the associations in table 2. Note that the association from *penguin* to *ice* is much stronger than the one from *ice* to *penguin*, as you would expect. This asymmetry would not be recovered with a mutual information measure.

cue word	**association**	**association strength**
penguin	*fish*	0.067
	ice	0.066
	sand	0.009
ice	*penguin*	0.010

Table 2: some association strengths calculated from co-occurrence statistics given by a web search engine.

This method is still quite coarse, since common words (e.g. *person*, *information*, *time*) will not only occur, but co-occur, in many documents (especially long documents), leading you to conclude that they have a strong association. A more refined measure would take into account how far apart the words are in the document. This can be done using a “sliding window” that lists a certain number (e.g. 10) of consecutive words in a document. If two words co-occur in this list, they are considered to be associated, otherwise not. The “window” moves word by word through the document, e.g. first listing words 1 to 10, then words 2 to 11, then 3 to 12, etc., until the last word of the document has been reached, after which the process starts anew with the next document. The same formula as above can be used except that *N* now stands for the number of windows, rather than the number of documents, in which a word or conjunction of words appears. If both the collection of documents and the window are large enough, then infrequent, but associated words, will still regularly appear together. For example, *penguin* and *Antarctica* will co-occur in phrases such as “penguins, who live in Antarctica” or “... like to visit Antarctica and see the penguins”. On the other hand, words that are common but not specifically associated, such as *person* and *information*, will not co-occur in a large number of windows relative to the total number of their occurrences, and therefore not get strong associations.

A remaining problem is to determine the optimal window size. When word co-occurrences are used to disambiguate meanings (e.g. the occurrence of *money* together with *bank* indicates that the latter should be interpreted in its "financial institution" sense), the best results seem to be produced when the window size is not too large [Ide & Véronis 1998], since words further apart from the target word may have no real association with it, and thus obscure the results. The problem is that different authors tend to find different optimal window sizes, which is understandable since there is no clear separation between the neighboring words that are strongly associated and those that are not. The method could be further refined by taking into account the variable distance between co-occurring words, so that words that appear closer together would receiver stronger associations than words that are farther apart. This has been applied e.g. by [Burgess 1998, Burgess et al. in press], who gave word co-occurrences a weight decreasing in step with the number of intervening words: e.g. adjacent words would get weight 5, words separated by 2 words weight 3, words separated by 5 or more words weight 0.

Inspired by our understanding of brain mechanisms, I would suggest a somewhat more sophisticated approach. A simple mathematical model would be to let the degree of association decrease *exponentially* (rather than linearly) with the number of separating words. For example, with one word separation half of the activation might remain, with two words a quarter, and with three words one eight. This can be motivated by the hypothesis that, for words held in short term memory, activation decays at a constant rate: with every time step, a fixed percentage of the activation is lost through diffusion, so that activation gradually "evaporates" until practically nothing is left. Thus, when a person hears the word *river* and two minutes later the word *bank* not enough memory of the former would be left to create a strong association with the latter, but a little association may still survive. To simplify computations, once weight has gone below a certain low threshold (e.g. 2%) it may be set to zero. This maintains a finite window size, even though the exponential decay approach allows us to use much larger windows without losing the reliability of associations.

It is worth noting that, in order to derive good associations from word co-occurrence, we really need the very large language samples that are found most easily on the web (10 million words as a minimum, according to [Rapp & Wettler 1991]). For example, the group of [Burgess 1998, Burgess et al. in press] used a 300 million word corpus gathered from Usenet discussions as input, to calculate all co-occurrence strengths between 70,000 distinct words. On the other hand, [Ide & Véronis 1998] observe that the traditionally used Brown corpus, which consists of one million words, is far too small to provide reliable co-occurrences: "in a window of five words to each side of the word *ash* in the Brown corpus, commonly associated words such as *fire, cigar, volcano*, etc. do not appear. The words *cigarette* and *tobacco* co-occur with *ash* only once, with the same frequency as [unrelated] words such as *room*, *bubble*, and *house*."

3.3. Deriving associations from user choices

Counting word co-occurrences in texts still provides only an indirect estimate of the associative meaning that people carry in their head. Although we cannot ask people to express all possible associations one-by-one, we might perhaps program a computer to learn these associations from their users in the same way that the users themselves have learned them, that is, by experience rather than by explicit instruction. Learning in the brain follows the rule of Hebb [Hebb 1967]: if two phenomena are experienced in close succession, the association between the concepts representing these phenomena is reinforced. If two associated concepts are not experienced together, their association gradually weakens. Thus, concepts that are frequently encountered together become strongly associated, while concepts that are rarely encountered together become weakly associated. If we look more closely at the underlying physiological mechanisms in the brain, we see that neurons ("concepts") that are activated ("experienced") in close succession develop a stronger synaptic connection ("association"). Moreover, this strengthening depends on the time interval between activation: short or zero intervals produce more reinforcement than longer intervals.

My co-worker Johan Bollen and I have used this principle to develop a self-learning web of associations between words [Bollen & Heylighen 1996, 1998; Heylighen 1999]. For our experiment, we selected the 150 most frequent nouns in the English language (according to the LOB corpus). To start up the associative network, we created a 150 x 150 matrix of cross-associations, initialized with small, random association strengths for each entry. For each word in the corpus, we then selected the 10 most strongly associated words, showing them to the users of the network in their order of strength. Thus, a user would see a page with a title word, such as *knowledge*, and a list of initially randomly chosen associated words, such as *trade, view, health, theory*, etc. (see first column of table 3). Out of those 10 words, the users were supposed to choose the one that is most strongly related to the title word *knowledge*, for example, *theory.* Selecting this word, brought them to a new page, now with *theory* as the title word, and a new list of 10 potentially related words, from which they were again supposed to choose the one most related to the title word. In that way, users would browse through a network of words linked by potential associations, each time selecting the link that seemed to best reflect the associative meaning of the title word. The network itself was made available on the World-Wide Web, so that people from anywhere in the world could participate in the experiment whenever they wanted.

The network was programmed to *learn* from the selections made by the users. The first learning rule, which we called "frequency", was the most obvious one: each time a link was chosen by a user, the corresponding association strength was increased relative to the other associations. Thus, frequently selected links would gather strong association values. Since linked words were ordered according to this strength, this means that words that were selected often would move up in the list of 10. However, this mechanism would only change the order of the list, not the choice of available words. Therefore, we introduced two additional learning rules. The most important of these rules, "transitivity", implemented the principle that links between concepts are strengthened even if there is a time interval between the successive activations. Thus, if a user would subsequently go from A to B, and from B to C, we not only strengthened the associations A → B and B → C, but also the indirect association A → C, albeit to a smaller degree. The idea is that if A (e.g. *knowledge*) is associated with B (e.g. *theory*), and B with C (e.g. *research*), then A is probably also somewhat associated with C.

This rule would now allow the network to strengthen associations between words that had no direct connections within the list of their 10 initial links. Since the initial random strengths were very small, any strength gained through the transitivity rule, even though it was smaller than the strength gained through the frequency rule, would be sufficient to make a word get ahead of the other words that had not been selected. Thus, the ordering according to strength might now add the new link *research* to the list of 10 most strongly associated words in *knowledge*, displacing the weakest link until then. The more people browsed through the network, the more potential new associations were created by transitivity. However, only the links that were really good associations would be directly chosen, and thus be rewarded by the frequency rule. The result was an evolutionary process of variation and selection, where promising candidates for new associations were constantly generated by transitivity, but only those that were good enough would receive sufficient reinforcement from the frequency rule to be maintained in the "top 10" list of associations.

The last rule, "symmetry", helped speed up the process, by suggesting additional candidates for good associations. The idea is simply that if there is an association from A to B, then probably there is also an association from B to A. When a user would go from word A to word B, the symmetry rule would not only reinforce the link A → B, but also the link B → A, albeit again to a smaller degree. This rule was only added in our second experiment. Although the first experiment using frequency and transitivity was surprisingly successful, the second experiment, with symmetry added, turned out to work even better, requiring about half the time to achieve a result of similar quality. Both experiments produced a rich associative network, in which all 150 words had gathered strong connections to the most related other words in the corpus. Table 2 illustrates the development of the list of 10 strongest associations for the word *knowledge.*

knowledge			
0	200	800	4000
trade	*education*	*education*	*education*
view	*experience*	*experience*	*experience*
health	*example*	*development*	*research*
theory	*theory*	*theory*	*development*
face	*training*	*research*	*mind*
book	*development*	*example*	*life*
line	*history*	*life*	*theory*
world	*view*	*training*	*training*
side	*situation*	*order*	*thought*
government	*work*	*effect*	*interest*

Table 3: self-organization of the list of 10 strongest links from the word “knowledge”, in different stages: initial random linking pattern, after 200 steps, after 800 steps, and after 4000 steps. A step corresponds to a user selecting a link on one of the 150 nodes, in a web that evolves according to the frequency, transitivity and symmetry learning rules.

The most remarkable thing about these experiments was how little effort was needed to develop a complex network of associations. With 150 words, the association matrix would contain 22,500 elements. Asking the volunteer participants to score each of those combinations with a numerical value would obviously have been totally impractical. Instead, the only things the participants did was to select one out of ten possible links, and this for as many words as they were willing to. On average, volunteers would go through about 10 words until they got bored. This means, that the average volunteer would make ten decisions, each time choosing one out of ten possibilities, a very small effort. Yet, we needed only about 2500 decisions in total, that is, about 250 participant sessions, for the associative network to achieve a fairly well-organized structure, in which most associated word-pairs had developed links reflecting their relative association strengths. In our papers presenting these experiments in detail [Bollen & Heylighen 1996, 1998], we discuss a number of mechanisms, such as positive feedback, that may explain the surprising efficiency of these learning web methods. Here, it may be sufficient to note that these methods are inspired by the functioning of our brain, which is obviously very good at learning associations.

After we analysed these experimental results, I have been reflecting about various more sophisticated learning rules that would make the network even more efficient. In particular, I propose to extend the transitivity rule in the following way: instead of only rewarding the indirect, two-step link A → C, we might reinforce *all* indirect links, A → D, A → E, A → F, etc., but with exponentially decreasing rewards. The rationale is the same mechanism of constant rate memory decay that I suggested to use when calculating exponentially decreasing weights for word co-occurrences. For example, if the reward for the two-step link A → C would be one half of the reward for the direct link A → B, then the three-step link A → D would get one quarter, A → E one eight, etc.

This would greatly increase the total number of rewards given for an average link selection. For example, for a 10 step sequence of link selections, it can be easily calculated that 53 links would receive rewards of varying degree. This would significantly accelerate the development of a differentiated matrix of associations. Although most additional rewards would be small, they might be sufficient to create direct links between words that were several steps removed from each other in the initially random network, thus providing a much larger variety of potential associations with the chance to be reinforced by the frequency rule. This would be especially useful for networks consisting of very large numbers of words.

There is still the question in how far the resulting link strengths offer a good measure of people's intuitive associations. One test is to check the correlation between the results of our experiment and word association norms, as [Spence & Owens 1990] did for co-occurrence. Unfortunately, given our quite limited list of 150 nouns out of which associations could be chosen, and the general sparseness of word association norms, there was little overlap between our data and the norms provided by [Nelson et al. 1998]. In those cases were there was some overlap, positive correlations were found, but this can hardly be considered a reliable validation.

To get more reliable evidence, my colleague J. Bollen has set up a smaller scale experiment using the 40 most frequent nouns in Dutch. This collection was small enough so that he could ask the experimental subjects (Flemish psychology students) to directly indicate which words were associated with which other words. Still, a 40 x 40 = 1600 list of potential associations is too large to be scored one by one. Therefore, the subjects were provided with a special graphical software which allowed them to see all 40 words on the computer screen, and draw connections between those words which they considered to be associated. The resulting graphical network can be viewed as representing the user's "mental map" of the words' associations. This map was converted to a simple matrix consisting of 1's (representing a connection) and 0's (representing the absence of connection). The matrix components for the different participants could then be averaged: e.g. if 10 out of 30 participants drew a connection from word *i* to word *j*, then the association a_{ij} would get strength 0.33. The resulting average matrix represents a "collective mental map" of all users [cf. Heylighen 1999]. When the averaged matrix and the association matrix derived from a learning web experiment were compared, it turned out that their components were strongly correlated, indicating that two very different methods to estimate intuitive word associations came to similar results.

4. Applications of associative networks

I have argued that with present-day computer and communication technologies, it is feasible to determine the associative meaning of words in the form of large matrices of associations. These matrices can be used to tackle various problems related to ambiguity in language. Let us now sketch some of these applications.

4.1. Classifying words and meanings

From an association matrix it is easy to derive a measure of the similarity between words. You could derive similarity directly from association strength, e.g. by taking the average of the association A → B, and B → A as the degree of similarity s(A, B) between A and B. This, however, may be misleading, as words can be strongly associated (e.g. *cradle* and *baby*, or *mouse* and *cheese*), yet have a very different meaning. A more reliable way is to measure similarity indirectly, as the inverse of the distance between the two vectors (a_i) and (b_i) that represent the two words' connotative meanings. This inverse distance is usually calculated as the normalized inner product, or cosine, between the two vectors:

$$s(A,B) = \frac{\sum_i a_i b_i}{\sqrt{\sum_i a_i^2}\sqrt{\sum_i b_i^2}}.$$

Defined in this way, s will vary between 1 (when the two vectors are identical or proportional) and 0 (when the two vectors are orthogonal). More generally, s will increase as A and B share more non-zero components a_i and b_i. Shared components mean that A and B have associations with the same other words. This means that A and B are to some degree interchangeable: in contexts where A occurs, B might occur just as well. This will be the case when A and B are either synonyms, e.g. *often* and *frequently*, or A and B represent distinct categories, but which share many features and tend to occur in the same circumstances, e.g. *ice* and *snow*, or *cat* and *dog*. Still, the fact that two synonyms are in principle interchangeable does not mean that you will find them to the same degree in the same circumstances. For example, *little* is more likely to be associated with *child* and less with *car*, while the opposite is true for its near-synonym *small*. Such subtle differences could never be expressed by looking at the denotation or extension of a word. Yet, association vectors do make a clear distinction between these two connotations.

Once you can determine a semantic similarity between words or other items, it becomes possible to cluster similar items together, in order to form larger categories. We have done that with the results from our web learning experiment. Applying a clustering algorithm to the association matrix allowed us to group most of the 150 most frequent nouns into 9 superclasses (see table 4).

"Time":	*age, time, century, day, evening, moment, period, week, year*
"Space":	*place, area, point, stage*
"Movement":	*action, change, movement, road, car*
"Control":	*authority, control, power, influence*
"Cognition":	*knowledge, fact, idea, thought, interest, book, course, development, doubt, education, example, experience, language, mind, name, word, problem, question, reason, research, result, school, side, situation, story, theory, training, use, voice*
"Intimacy":	*love, family, house, peace, father, friend, girl, hand, body, face, head, figure, heart, church, kind, mother, woman, music, bed, wife*
"Vitality":	*boy, man, life, health*
"Society":	*society, state, town, commonwealth*
"Office":	*building, office, work, room*

Table 4: clusters of words derived from the learning web experiment [Bollen & Heylighen 1996, 1998].

[Burgess 1998] performed a more limited clustering with some of the words of his huge co-occurrence database. It turned out that the associative network not only allowed a discrimination between semantic categories (e.g. animals and people), but also between grammatical categories, such as nouns and verbs, past tense verbs and past participles, or demonstrative and genitive determiners. This can be understood from the fact that Burgess's co-occurrence calculations take into account the distance between words in a sentence. Since different grammatical categories tend to be at specific distances from other grammatical categories (e.g. an article can be followed by a noun or an adjective, not by a verb or another article), co-occurrence data therefore implicitly contain information about grammar and word order. This may be sufficient for a computer—or a child—to autonomously learn syntactical categories [cf. Finch & Chater 1992].

Such superclasses correspond to broad, abstract categories. However, clustering can also be used to discriminate between subclasses of more specialized meanings. Consider an ambiguous word that has more than one meaning, such as *bank*. Some of the associations of *bank* with other words will depend on one meaning, e.g. *money*, *interest*, while other associations depend on its second meaning, e.g. *river* or *edge.* If you now use the list of all words associated with *bank* and perform a cluster analysis you will find clearly distinct clusters corresponding to the distinct meanings. If you then take the vector of all associations that *bank* has with other words, then you can split it up into two vectors, respectively using the components of the one and of the other cluster (components that belong to both or to neither cluster can be evenly divided, or divided proportionally to the distance they have with either cluster). These two vectors, whose sum again forms the original *bank* vector, now represent the two distinct meanings: "bank as financial institution" and "bank as river edge". This methodology can be used even if a word has several meanings and if those meanings are very close, as demonstrated by [Schütze 1998] on the basis of co-occurrence data.

A sufficiently smart clustering algorithm should thus be able to tell you how many meanings a given word has, and provide an intuitive characterization of each meaning by labelling it with the strongest components in its cluster (e.g. *bank: money, interest, rate*; and *bank: river, sand, border*). A dictionary or thesaurus of meanings, such as WordNet [Miller et al. 1990], could then be used to look up the best corresponding meanings (by hand or automatically, by computing similarities between meaning vectors and the words used in the thesaurus' description), thus providing a more explicit label (e.g. *bank: financial institution*, and *bank: edge of river*) [cf. Ide & Véronis 1998].

After discriminating between meanings or senses, the next logical step is to determine which of those meanings should be used in a given context.

4.2. Spreading activation

Let us go back to our example of the "seagulls sitting on the bank”. When people read that sentence, they have little difficulty interpreting *bank* as "river edge". The reason is that hearing the word *seagull*, because of its strong association with watery surfaces (cf. Table 1), already creates an expectation of something to do with water. This expectation afterwards facilitates the interpretation of *bank* in its water-associated meaning. The underlying mechanism has been experimentally investigated through the phenomenon of *semantic priming* [cf. Burgess 1998; Lowe 1997]. This is investigated with the following psychological experiment: subjects are shown or hear a first word (e.g. *seagull*), the "prime", followed shortly by a second word (e.g. *water*), the "target". They then have to perform as quickly as possible some action, e.g. push a button, to show that they have understood the target word. The recurrent observation in these experiments is that if the prime is associated with the target, then the subjects react more quickly. In other words, having encountered a semantically related word makes it easier to recognize a subsequent word. The effect does not exist with semantically unrelated words, such as *seagull* and *money*: encountering the first word will not influence the speed with which you recognize the second word.

The standard interpretation of this phenomenon is the following [Lowe 1997]: recognizing the prime word activates that part of your memory or brain that corresponds to that word's meaning. This activation tends to diffuse or spread towards associated meanings. If then a word corresponding to one of these associated meanings is to be recognized, the already present low level of activation makes it easier to activate that meaning fully, and thus consciously interpret the word. [Burgess 1998] has shown that the semantic priming effect can be simulated with an associative matrix derived from word co-occurrence: the degree to which one word can prime another word in a psychological experiment is correlated with the degree of semantic similarity between word vectors.

In semantic priming experiments, activation seems to spread from a single prime word to a single target word. In a more realistic situation (e.g. when reading a text), on the other hand, several words will be activated, and this activation will spread in parallel to a large group of associated words. This "parallel" spreading can be conceived in the following way. Assume that two words, A and B, both are associated with a third word, C. Activation of either A or B will bring a little activation to C. When both A and B are activated, on the other hand, activation will enter C from both sides and the total activation of C will be the sum of the activations coming from A and from B. The amount of activation entering C will moreover depend on the degree of activation of A and B and on the strength of the associative links from A and B to C. This can be easily represented in our vector space model of word associations. The "input" vector represents the initial degree of activation of various words. If a vector component is zero, this means that the corresponding word is not activated. If it is 1, it means the word is fully activated. This vector can now be multiplied with the association matrix to represent a single step of "spreading". This ensures that all words that have a non-zero association with an initially activated word receive an amount of association from it proportional to the strength of their incoming associative link from that word.

This multiplication can be repeated several times to represent a longer, multi-step process of spreading activation. Multi-step spreading seems like a more realistic mechanism, since the brain remains active constantly and does not stop after a single step. Moreover, it allows us to explain phenomena such as mediated priming, where a prime word (e.g. *lion*) can facilitate the recognition of a target word (e.g. *stripes*) even though they are only linked indirectly, by their shared association with a third word (e.g. *tiger*). [Rapp & Wettler 1991] have moreover shown that multi-step processes work better than single-step ones when trying to model word associations with co-occurrences.

We have implemented such a multi-step spreading activation in the associative network derived from our learning web experiment. If, for example, the words *control* and *society* are initially activated, this activation spread to other words, with the highest amount of activation ending up in the word *government*. Similarly, if *work*, *room* and *paper* are activated, the word that receives the highest activation was *office.* Note that this is a way of retrieving concepts by indirect characterization. Neither *work*, *room*, nor *paper* on their own would make you immediately think of *office.* Yet, when all three are simultaneously present, the word *office* seems like the most obvious thing that connects them all together.

Although he did not use spreading activation in his network, [Burgess 1998] demonstrated a similar effect with the data from his co-occurrence analysis: when he characterized a word by the list of words that were most close to it in the semantic vector space, then people could often guess what the word was. This may remind us of a classic memory experiment in which subjects listen to a long list of words (e.g. *cool, ice, hot, freezing, winter, ...*) that are all closely associated to a target word (e.g. *cold*). They are then asked to recall as many of the words they heard as possible. If the associated words are well-chosen, subjects will typically recall the target word, even though they did not hear it! The explanation, again, is spreading activation: the target word has received so much activation from its surrounding associates that it becomes as strongly activated (or even more strongly) as any of the actually perceived words.

4.3. Resolving ambiguity

How can we use spreading activation to resolve ambiguity? The most straightforward approach may be to activate all the words in a sentence that have already been recognized (e.g. *seagulls, sitting, on, the*) and then let activation spread from those in order to prepare the ground for the interpretation of the following word (e.g. *bank*). Since a word like *the* has no very strong associations, its activation will be distributed more or less evenly in all directions, contributing very little to the activation of any particular word. (This observation is often used to simplify the model by ignoring all such "function" words, like articles and prepositions, that contain little information). A more specific word like *seagull*, on the other hand, will have a limited number of strong associations, to words such as *water* (cf. table 1), along which a lot of activation will flow. From *water*, activation may spread further to other words such as *river*. If then the word *bank* is perceived, with one meaning strongly associated with *river*, then it is this meaning, rather than the meaning "financial institution" that will receive most activation, and therefore be selected as best interpretation.

One way to implement this is to calculate the similarity between the vector representing the activation that has spread from the preceding—and possibly subsequent—words (the "context") and the vectors that represent the different senses of the word that still needs to be interpreted. The sense vector that is most similar to the context vector can then be selected as best interpretation. A simple version of this algorithm, with only a single spreading activation step, was used successfully by [Kikui 1999] to resolve ambiguity in English-Japanese translations.

A somewhat more involved method is to create inhibitory links between the different, mutually exclusive senses of the word. This means that whenever one sense receives activation, the activation in the other senses is proportionately inhibited or decreased. Thus, the different senses are forced to "compete" for the available activation during the successive spreading steps, and the one that receives most can suppress activation in its rivals. [Véronis & Ide 1995] have shown that such a spreading activation process can efficiently disambiguate word senses (although their work was limited by the small number of associations they could derive from co-occurrence in dictionary definitions, rather than from large-scale web processing).

Let us consider in more detail how to specify the initial activation vector. Activating only the words in the preceding phrase seems rather unreliable, since the context element that disambiguates a particular word may have occurred well before the present phrase. On the other hand, if we would evenly activate all preceding words in the text, we run the risk that activation would become so diffuse that any power of discrimination is lost. Again, the most logical solution may be to activate all preceding words, but in such a way that activation decreases exponentially with the distance to the target word.

4.4. Information retrieval in the web

An at first sight very different application of these principles is the choice between documents rather than between word meanings. The idea is that a user browsing through the web will be looking for those documents that best match his or her interest. This "interest" is actually similar to the word meanings which we have discussed until now. For example, a user may be looking for information on "banks as financial institutions". The traditional way to do this is to use a search engine, enter the keyword *bank*, and let the search engine select all documents that frequently use the word *bank*. This brings about several problems [Heylighen 1999]. Most obviously, the search engine does not distinguish between "bank as financial institution" and "bank as river edge", and therefore will return documents on either subject. [Schütze 1998] has shown that the kind of word disambiguation methods which we discussed can make it easier for the user to find the desired documents. For example, when the search engine encounters an ambiguous word, it may reply with a list of possible meanings for that word (represented, e.g. as clusters of associated words), and let the user select the most appropriate one. It can then use the words most strongly associated with each meaning (e.g. *money, rate,* vs. *river, edge*) to classify documents containing the word *bank* according to whether they are closer to the one or to the other meaning, and only return the documents that match the desired meaning.

A more serious problem is that the interest of the user in general cannot be defined by one or a few keywords. For example, the user may be interested in documents that discuss the banking business in a light-hearted, playful manner. There is no obvious way to select that kind of documents by entering keywords. Which are the words that light-hearted documents would particularly use? The only way to establish that a document is light-hearted is to let it be read by another user, who would be able to intuitively estimate its light-heartedness. But even if we would have a committee of users that would establish the degree of light-heartedness of all documents on the web, allowing us to attach "light-heartedness" labels to specific documents, this would still only help a fraction of all the different users that are looking for different styles and types of documents. Moreover, the user searching for documents on banking may not even know that she is looking for light-hearted texts, but would still very much prefer to read such a text if she could find one. In that case, the label would be of little use.

More generally, the user's interests and preferences may depend on a myriad of factors, most of which she would not be able to formulate explicitly. Yet, if she would find a page satisfying those interests, she would intuitively recognize it. This is similar to the problem of meaning: in general it is very difficult, if not impossible, to explicitly define or formulate the full meaning of a word or phrase; yet, we have no difficulty grasping that meaning when we encounter it. Therefore, we might try to tackle the problem of modelling interest in the same way as we tackled the problem of modelling meaning: by creating a network of associations and exploring it through spreading activation.

The methodology that we used to create a learning web of associations between word can be applied straightforwardly to create a learning web of associations between documents. It suffices to reinforce direct or indirect links followed by users "surfing" from page to page. A strong link between two pages A and B then would mean that most users who looked at page A, also looked at page B. To reliably model an association of interest, one more element needs to be added to the algorithm. Since users browse the web by selecting links rather than selecting documents, it is in principle possible that many users would "click" on the link pointing to document B because it *looks* interesting, but then would discover that the document does not fulfil its promises. In that case, the present learning web algorithm would still reward the link to B, thus persistently drawing readers to a disappointing page. In order to avoid this, the algorithm must take into account some implicit or explicit evaluation of the interestingness of a document.

A simple, implicit measure is the duration of a visit: the more interesting the page, the more time the user will spend reading that page. It has been shown empirically that this measure gives a good estimate of the interest for a document rather than for its length [Nichols 1998]: users do not spend more time with long, irrelevant documents than with short, irrelevant documents. Still, the degree of interest is not strictly proportional to the time spent in front of the page: there can be any number of reasons, such as an incoming phone call or a visit to the coffee room, why a user might keep a document open on his or her web browser for an extended period of time. If the document is clearly irrelevant, on the other hand, the user is likely to immediately click on a further link, or go back to a previously consulted document. To take these effects into account, we need a formula that extracts the most significant part of the measurable duration between page visits. It is possible to define a sigmoid-like function that reduces all very short intervals (up to about 5 seconds) to zero, then increases almost linearly with time, slows down and reaches a horizontal asymptote after a few minutes. The precise parameters of the function will need to be tuned by experimentation, but the approach should be clear: only the interval between first viewing the page and reading several paragraphs of it is really significant as an estimate for the interest a user finds in that page.

Whether we use this implicit estimate based on duration, or an explicit evaluation by the user, once we know how interesting the user found a page, we can produce a more reliable formula to calculate association strength. Instead of rewarding a link A → B with a fixed amount, we can reward it with an amount proportional to the degree of interest the user has for both A and B. Thus, links between uninteresting pages will receive little reinforcement, while links between interesting pages will receive a lot. Another way to see this is to consider interest expressed by a user (e.g. through the time spent reading the page) as a form of "activation" received by a virtual "neuron" that represents the page. If two neurons are subsequently activated, then the rule of Hebb tells us that the synapse linking them should be reinforced, in proportion to their degree of activation.

The longer the time that has passed between the subsequent activations, the weaker the reinforcement should be. I suggested earlier that reinforcement should decay exponentially with the number of intermediate words or page visits. In an environment where the time spent reading (or drinking coffee) can vary greatly, it seems more accurate to let reinforcement decrease exponentially with the continuous *duration,* rather than the discrete *number of steps,* in between page visits. Again, this can be motivated by a general memory model in which activation decays at a constant rate: the longer the time that has passed since a user concentrated on a particular topic, the more likely it is that his or her focus of attention has shifted to a different domain. Thus, if a user has been reading page A and, two hours later, page B, then A and B most likely have nothing particular in common. On the other hand, if the user attentively reads B two minutes after reading A, then A and B probably treat closely related subjects.

To build an associative network based on these observations, it suffices to mine data from a website's log [Bollen, Van de Sompel & Rocha 1999]. Such a log lists which pages have been consulted by which user at which moment. This is enough to reconstruct the complete path of pages a given user has been subsequently browsing during one session, as well as the duration spent browsing each individual page (out of which the page's "activation" can be computed), and the duration in between two different page visits (out of which the exponential decay factor can be computed). This gives us all the information we need to provide accurate reinforcements to all the potential links between pages, and thus compute their overall relative strength.

How can we use the resulting network of associations to facilitate information retrieval in the web? First, as in our original web of word associations, learned links would be appended to the web page in the order of their strength. Thus, visitors to a given page would immediately get a list of recommended pages in the order of their estimated relevance. "Relevance" here simply means "interestingness, relative to the subject of the present page". For example, a high quality page on the history of relativity theory might provide a link to another high quality page with a biography of

Albert Einstein, originator of the theory. The link would mean that people interested in the first page would most likely also be interested in the second page. This implies two things: 1) pages would get clustered according to the degree of similarity in their subjects. Pages on widely different subjects, e.g. relativity theory and football, would only have very weak connections; 2) in any given domain, high quality pages would get more and stronger incoming links than low quality pages. This means that if you are interested in a particular subject, it would suffice to find any page on that subject (e.g. using a traditional search engine or subject index), in order to be immediately led to strongly related pages, allowing you to explore the domain in the most efficient way.

Creating relevant links is only the most rudimentary application of associative webs, though. Another relatively straightforward application is the clustering of web pages according to their associative similarity, so as to create automatic indexes of pages on the same topic, or a classification of subjects. A more sophisticated application is spreading activation from a "context vector", as discussed in the preceding section. A person subsequently reading documents about a particular subject on the web is in a way similar to a person subsequently reading words which together form a coherent sentence or discourse. We have discussed how associative network technology might automatize or support that process: the context of preceding words can "activate" a cluster of meaning in the network, helping us to guess the correct sense for the next word.

Let us apply the same mechanism to web browsing: each document a user reads receives activation proportional to the interest shown by that user (e.g. as derived from the reading duration). This activation decays exponentially: the more time has passed since the user read a document, the less of its activation remains. All documents are connected by a huge network of associations, learned from previous user activities, and represented by an associative matrix. This network can now be used to let the activation spread to related pages. From those, the as yet unread pages are recommended to the user, in the order of their degree of activation. Each time the user starts reading a page, this page too is considered activated in proportion to the user's appraisal of it, and a new spreading activation process is carried out, producing an updated list of recommendations. Thus, wherever the user goes, the associative network accompanies and supports him or her with an increasingly reliable list of recommendations, tailored to the user's specific interests of the moment, but taking into account his or her general preferences as implicitly expressed through earlier choices. In that way, the web becomes an intelligent, intuitive companion, that anticipates every user's wishes and desires.

5. Conclusion

This paper has reviewed the basic principles underlying a number of still experimental—or even speculative—computer technologies. These technologies distinguish themselves by representing knowledge, interest and meaning by means of a network of associations. These associations may connect words, concepts, or documents. However, it is not the components that are connected, but the connections themselves that play the central role. An component in such a network only gets its meaning through its relations with other elements. As such, these networks are "bootstrapping": their components mutually support each other [Heylighen, in press]. This avoids many of the deep problems, such as "symbol grounding", that until now have plagued more traditional knowledge representations in Artificial Intelligence. Moreover, these networks are intrinsically "soft" structures, that can easily shift or adapt to tiny changes in associative strength. This allows them to express all the vague, fleeting, intuitive, context-dependent meanings that are otherwise so difficult to grasp [cf. Heylighen 1991; Heylighen & Dewaele, in press].

The disadvantage of associative networks is that they tend to be huge. For example, a typical word co-occurrence matrix as created by the group of [Burgess 1998, Burgess et al. in press] contains 140,000 by 140,000 numbers. Most researchers have therefore reduced the dimensionality of their matrices, using methods such as multidimensional scaling, principal components analysis, singular value decomposition, or simply selecting only the most variable or informative context words to compute associations with. This simplifies processing, allows low-dimensional, graphical representations of the semantic space [e.g. Schütze 1998; Lowe 1997], and may even improve the quality of the model by removing "noise" from the data, as exploited by Latent Semantic Indexing techniques [Deerweester et al 1990; Foltz 1996]. Yet, present computers and algorithmic techniques seem powerful enough to work even with the full matrices, and this opens up a whole new range of possible applications that still need to be explored.

The more profound problem is the knowledge acquisition bottleneck: how do you transfer such a huge amount of associations from a person's mind into a computer? A solution is suggested by the idea that originally inspired the use of associative networks: their similarity to the organization of the brain, with its variable strength synapses connecting neurons. In the brain, associations are learned through the rule of Hebb [1967]: phenomena that are encountered in close succession become more strongly associated. A similar mechanism should allow a computer too to learn associations. The problem is still that in order to extract a rich network of associations the computer must receive a sufficiently large input of data.

This bottleneck can be overcome with the help of the world-wide web. Previous computer-aided attempts to analyse language had to rely on manually entered corpora, which very much limits their size and representativeness. The present web, on the other hand, provides automatic and free electronic access to billions of natural language samples, scattered over the most diverse subject, genres and styles. The most obvious method to mine these riches for their implicit associations is co-occurrence measurement: counting the number of times a particular word appears in the neighborhood of another word. These co-occurrence frequencies have been shown to provide good indirect estimates of associative strength.

The most direct method would let people estimate the relative association between words, documents or items. Although it is impossible to have all possible associations scored individually, my colleagues and I have developed a somewhat less direct method, *web learning*, that seems to work surprisingly well. The idea, inspired by a generalized rule of Hebb, is that the more users actively use direct or indirect connections, the stronger the corresponding links should become. Thus, the network learns from the collective activity of its users. Like for the co-occurrence approach, I have proposed to improve the method by making reinforcements decrease exponentially with the time interval between subsequent encounters. Such a method could in principle turn the web itself into a huge, associative memory connecting all possible subjects.

Whichever method is used to develop it, once an associative network is available it can be applied to tackle various fundamental tasks required for language technologies, such as clustering of words into categories, discrimination of word senses, and disambiguation on the basis of context. The underlying process is equivalent to the mechanism of spreading activation that is assumed to underlie all cognitive processes in the brain: elements activated by a given context in their turn activate a neighborhood of other, closely associated elements. Thus, the associative network becomes more than a static memory: it becomes an active, brain-like processor, capable of interpreting input, choosing between alternatives, or even "thinking ahead". If the web itself would become such an active associative network, we might truly call it a "global brain": a world-wide intelligent network that constantly learns from its users, and helps them by anticipating and finding answers to all of their questions [Heylighen & Bollen 1996; Heylighen 1999].

References

American Heritage Dictionary of the English Language (1996) Third Edition. Houghton Mifflin Company.

Bar-Hillel, Yehoshua (1960) Automatic Translation of Languages. In Alt, Franz; Booth, A. Donald and Meagher, R. E. (Eds), *Advances in Computers*, Academic Press, New York. 247-261.

Bollen, Johan & Heylighen, Francis (1996) Algorithms for the Self-organisation of Distributed, Multi-user Networks. Possible application for the future World Wide Web, in *Cybernetics and Systems '96* R. Trappl (ed.), Austrian Society for Cybernetics, Vienna, 911-916.

Bollen, Johan & Heylighen, Francis (1998) A system to restructure hypertext networks into valid user models, *New Review of HyperMedia and Multimedia*, 189-213.

Bollen, Johan, Herbert Van de Sompel, and Luis M. Rocha (1999) Mining associative relations from website logs and their application to context-dependent retrieval using spreading activation, in *Workshop on Organizing Web Space* (WOWS), ACM Digital Libraries 99, August 1999, Berkeley, California.

Burgess, C. (1998) From simple associations to the building blocks of language: Modeling meaning in memory with the HAL model. *Behavior Research Methods, Instruments, & Computers*, 30, 188-198.

Burgess, C., Livesay, K., & Lund, K. (in press) Explorations in Context Space: Words, Sentences, Discourse. *Discourse Processes*, 25, 211 - 257.

Cairns, P., Shillcock, R.C., Chater, N., Levy, J. (1997) Bootstrapping word boundaries: a bottom-up corpus-based approach to speech segmentation. *Cognitive Psychology*, 33, 111–153

Church, K. W. & Hanks, P. (1990) Word Association Norms, Mutual Information, and Lexicography. *Computational Linguistics* 16, 22-29.

Deerweester S., Dumais S., Landauer T., Furnas G. and Harshman R. (1990) Indexing by Latent Semantic Analysis, *Journal of the American Society for Information Science* 416, 391-407.

Finch, S. and N. Chater (1992) Bootstrapping Syntactic Categories Using Statistical Methods. In *Proc. 1st SHOE Workshop*.. Tilburg University, The Netherlands, 229-235.

Foltz, P.W. (1996) Latent Semantic Analysis for text-based research, *Behavior Research Methods, Instruments, & Computers*, 28, 197-202.

Hebb, D. O. 1967 The organisation of behavior: a neuropsychological theory. Science Editions, New York.

Heylighen, Francis (1991) Design of a Hypermedia Interface Translating between Associative and Formal Representations, *International Journal of Man-Machine Studies* 35, 491-515.

Heylighen, Francis (1999) Collective Intelligence and its Implementation on the Web: algorithms to develop a collective mental map, *Computational and Mathematical Theory of Organizations* 5(3), 253-280.

Heylighen, Francis (in press) Bootstrapping knowledge representations: from entailment meshes via semantic nets to learning webs, *Kybernetes* **.**

Heylighen, Francis & Bollen, Johan (1996) The World-Wide Web as a Super-Brain: from metaphor to model, in *Cybernetics and Systems '96* R. Trappl (ed.), Austrian Society for Cybernetics, Vienna, 917-922.

Heylighen, Francis & Dewaele, Jean-Marc (in press) Variation in the contextuality of language: an empirical measure, *Foundations of Science* **.**

Ide, Nancy and Véronis, Jean (1998) Word sense disambiguation: The state of the art. *Computational Linguistics*, 241, 1-40.

Kikui, Genichiro (1999) Resolving Translation Ambiguity using Non-parallel Bilingual Corpora. In *Proceedings of ACL99 Workshop on Unsupervised Learning in Natural Language Processing*.

Lenat, Douglas B. and Guha, Ramanathan V., (1990) *Building large knowledge-based systems*. Addison-Wesley, Reading, Massachusetts.

Lowe, Will (1997) Semantic representation and priming in a self-organizing lexicon, in *Proceedings of the 4th Neural Computation and Psychology Workshop,* Springer Verlag, 227-239

Miller, George A., Beckwith, Richard T., Fellbaum, Christiane D., Gross, Derek, and Miller, Katherine J. (1990) WordNet: An on-line lexical database. *International Journal of Lexicography,* **3**(4), 235-244.

Moss, H., Older, L. (1996) *Birkbeck Word Association Norms*, Lawrence Erlbaum Associates Hove, UK

Nelson, D. L., McEvoy, C. L., & Schreiber, T. A. (1998) *The University of South Florida word association, rhyme, and word fragment norms*. http//w3.usf.edu/FreeAssociation/.

Nichols, D.M. (1998) Implicit Rating and Filtering, *Proc. Fifth DELOS Workshop on Filtering and Collaborative Filtering*, Budapest, Hungary, 10-12 November 1997, ERCIM, 31-36.

Palermo, David S., & Jenkins, James J. (1964) *Word Association Norms, Grade School Through College*. University of Minnesota Press, Minneapolis.

Rapp, R. & Wettler, M. (1991). Prediction of free word associations based on Hebbian learning. in *Proceedings of the International Joint Conference on Neural Networks,* Singapore, Vol.1, 25-29.

Schütze, Hinrich (1998) Automatic Word Sense Discrimination, *Computational Linguistics* 24, 1, 97-124.

Spence, D.P. & Owens K.C. (1990) Lexical co-occurrence and association strength, *Journal of Psycholinguistic Research* 19, 317-330.

Véronis, Jean & Ide, Nancy (1995) Large Neural Networks for the Resolution of Lexical Ambiguity. In Saint-Dizier, P., Viegas, E. (Eds.) *Computational Lexical Semantics*. Natural Language Processing Series, Cambridge University Press, 251-269.

Wettler, M. & Rapp, R. (1993) Computation of word associations based on the co-occurrences of words in large corpora. In *Proceedings of the 1st Workshop on Very Large Corpora: Academic and Industrial Perspectives*, Columbus, Ohio, 84-93.