INFORMATION COMPRESSION BY MULTIPLE ALIGNMENT, UNIFICATION AND SEARCH AS A MODEL OF NON-CONSCIOUS INTELLIGENCE*

J Gerard Wolff
School of Informatics, University of Wales Bangor, Dean Street, Bangor, LL57 1UT, UK.
gerry@sees.bangor.ac.uk

April 18, 2001

Abstract

This paper presents the idea that several aspects of non-conscious intelligence may be understood as *information compression by multiple alignment, unification and search* (ICMAUS). The meanings of these terms are explained. The paper shows, with examples, how the ICMAUS framework can be applied to such aspects of non-conscious intelligence as natural language processing, best-match pattern recognition and information retrieval (including recognition through multiple levels of abstraction), probabilistic reasoning, and unsupervised inductive learning. Most of the examples in the paper are illustrated with output from SP61, a software model that is a partial realisation of the ICMAUS framework.

1 Introduction

It is widely acknowledged that many of the 'intelligent' things that people can do—learning a first language, understanding and producing language, probabilistic reasoning, recognising objects and patterns, recalling information, and others—are achieved non-consciously in the sense that we have little or no conscious insight into what we are doing. It is evident, for example, that using a natural language means knowing a complex set of rules but it is notoriously difficult to make these rules explicit and accessible to conscious consideration. And the processes by which young children learn these rules (or the processes of 'acquisition' if you favour a 'nativist' theory) seem also to be singularly free of conscious insights.

This paper presents, in outline, informally, and by means of examples, a conceptual and computational framework that seems to have potential to provide a relatively simple, coherent account of several of these aspects of human intelligence.

Although the framework seems to provide a useful account of several aspects of non-conscious intelligence, it was not developed primarily for that purpose. As outlined below, the intended scope of the theory includes aspects of 'computing', mathematics, logic and related disciplines.

*Presented at, and in the Proceedings of, the AISB'01 Symposium on Nonconscious Intelligence: From Natural to Artificial, March 2001, York, pp 72–81.

2 Background and Context

The proposals described in this paper grew out of a programme of research developing computer models of first-language learning by children (Wolff, 1988, 1982, 1980, 1977, 1975). An important insight emerging from this research was that many aspects of first-language learning may usefully be understood in terms of principles of Minimum Length Encoding (MLE) pioneered by Solomonoff, Wallace and Boulton, Rissanen and others (see (Li and Vitányi, 1997)).

The key idea here is that, in the inductive learning of a grammar or comparable knowledge structure, one should seek to minimise $(G+E)$ where G is the size of the grammar (in bits or comparable measure of information) and E is the size of the sample used to induce the grammar, after it has been compressed by encoding in terms of the structures specified in the grammar. In effect, MLE means information compression (IC), taking into account the information 'cost' of the coding system (grammar) required to achieve IC.

IC is achieved by the removal of *redundancy* from information, with or without the removal of non-redundant information too. Since redundancy often takes the form of repeating patterns in information, IC can often be achieved by *looking for patterns that match each other* and then merging or *unifying* patterns that are the same. Given that, typically, there is a large number of alternative ways in which patterns can be matched and unified, some kind of *search* is required to find those unifications that yield relatively large amounts of IC. Given that search spaces are often astronomically large, some kind of *constraint* on searching is required (using heuristic techniques or otherwise) to keep within the bounds of practicality.

Pattern matching, unification and search (PMUS) are prominent in all the 'standard' methods for IC and also in my MK10 model of how words and other segments in language are learned (Wolff, 1975, 1977, 1980) and in my SNPR model of syntax learning (Wolff, 1982, 1988). It became apparent gradually that PMUS seemed to be at the heart of a wide range of kinds of artificial 'computing' and also in aspects of human cognition other than language learning. These insights led to the hypothesis that many aspects of both 'computing' and 'cognition' might usefully be understood as IC by PMUS.

Since 1988, this hypothesis has driven a programme of research seeking to develop a conceptual framework dedicated to PMUS that might serve to integrate a range of concepts and phenomena in computing and cognition. The challenge has been to develop a framework combining *simplicity* with explanatory or descriptive *power* in several different areas of computing and cognition.

2.1 Multiple Alignment

A key insight emerging from this research is that the generality of the framework could be greatly enhanced by incorporating a concept of *multiple alignment* similar to but not exactly the same as the concept of multiple alignment in bio-informatics. In that context, a multiple alignment is an arrangement, one above the other, of two or more sequences of amino-acid residues or DNA bases so that, by judicious 'stretching' of selected sequences, symbols that match each other from one sequence to another are arranged in vertical columns. A 'good' alignment is one with a relatively large number of matches between symbols. As we shall see, a multiple alignment in the ICMAUS framework differs mainly in that one of the sequences is designated 'New', the remainder are 'Old', and a 'good' alignment is one that allows New to be encoded economically in terms of the sequences in Old.

3 Information Compression by Multiple Alignment, Unification and Search (ICMAUS)

In its most general form, the ICMAUS framework works like this:

- Each brain or computer receives 'raw' data from the environment, designated 'New'. This information may be analogue or digital but we shall assume that, at an early stage in processing, analogue information is converted into digital form. In the framework, all New information takes the form of arrays or *patterns* of atomic *symbols*. These patterns may have one or more dimensions but, in current work, attention is focussed mainly on one-dimensional sequences of symbols. In general, each symbol in the system is simply a 'mark' that can be matched in a yes/no manner with other symbols but otherwise has no intrinsic meaning. In current computer models of the framework, the assertion just made needs to be qualified because some symbols are classified as 'data' symbols and others are classified as 'code' symbols (more about this below).

- When each portion of New has been received, it is stored either as it is or in compressed form (described below) and changes its status to that of 'Old' information.

- As New information is received, the system tries to compress it as much as possible by searching for matching patterns between New and Old. Amongst alternative possible matches, the best (in terms of IC) are chosen and the corresponding patterns are unified. As an example, let us suppose that Old contained the pattern 'information compression' together with 'IC', a code symbol for that pattern. If New contains the pattern 'information compression' this may be unified with the pattern stored in Old and, in effect, deleted from New. The code symbol 'IC' will serve to represent 'information compression' in the encoded form of New. Those portions of New that cannot be encoded in this way are simply stored in Old 'as is'. Otherwise, the encoded form of New is stored.

- A feature of the framework not yet mentioned is that recognition is recursive: an initial encoding in terms of 'low level' patterns may itself be encoded in terms of 'higher level' patterns, and so on as long as there are patterns of redundancy that can be found. This is illustrated in our first example—parsing of natural language—described in Section 4, below.

- It is envisaged that, where storage space is limited, Old will be periodically purged of patterns that are not proving 'useful' in terms of IC. It is also possible that information that is not deemed to be 'useful' in other senses may be discarded at an early stage. Where storage space is not restricted, it is possible that less useful patterns are retained but are shunted into regions that are less accessible.

The overall framework just outlined combines 'recognition' with 'learning': if New information can be encoded in terms of Old information, this is 'recognition'. 'Learning' occurs when New information is stored, either in raw form or encoded in terms of Old. As we shall see, other interesting capabilities emerge as by-products of the framework. These include such things as information retrieval and probabilistic inference.

3.1 The SP61 Model

The recognition part of the framework (including the recursive aspect of recognition) is implemented in a software model, SP61. Most of the examples described below are illustrated with output from the SP61 model. The most detailed description of this model to date may be found in (Wolff, 2000). Work is currently in progress developing SP70, a 'full' model of the framework, including unsupervised inductive learning.

At the heart of the SP61 model is a process for finding 'good' full or partial matches between two one-dimensional patterns. This process is similar to 'dynamic programming' but with advantages compared with standard methods: it can find good matches between arbitrarily long patterns, it can deliver a set of alternative alignments, and the thoroughness of searching can be varied by varying the amount of memory available to store partial results.

The 'dynamic programming' process for finding good full or partial matches between two patterns is applied recursively so that 'multiple' alignments with arbitrarily many levels can be built up, as will be seen below.

For any given set of patterns in New and Old, the model can deliver a set of alternative alignments, graded in terms of IC. The measure of IC that is used is the compression of New that can be achieved by encoding it in terms of the patterns from Old that appear in the alignment. Details of the method are described in (Wolff, 2000).

3.2 Symbolic and Connectionist Processing

Although the examples to be presented have a 'symbolic' flavour, the principles embodied in the ICMAUS framework are more abstract than the distinction between 'symbolic' and 'connectionist' processing. The SP61 model has been implemented in a conventional manner using an ordinary programming language. However, it seems entirely possible that the entire framework could be implemented using connectionist mechanisms. This is something to be examined in the future.

3.3 Computing, Mathematics, Logic and Related Disciplines

Apart from aspects of non-conscious intelligence that will be considered below, the ICMAUS framework has been applied in an interpretation of concepts of 'computing' such as the *Universal Turing Machine* and the *Post Canonical System* (Wolff, 1999b). It seems also to provide a useful perspective on forms and structures in mathematics, logic and related disciplines and on inferential processes of calculation and deduction in those disciplines (Wolff, 2001).

4 Natural Language Processing

As noted in the Introduction, our ability to understand and produce natural language is singularly opaque to conscious insights and is thus a good example of non-conscious intelligence. It makes no difference whether the language is spoken or written.

Although no attempt has yet been made to apply the ICMAUS concepts to the 'semantic' aspects of NL analysis and production, the framework seems to accommodate NL parsing and syntactically-driven language production, and there are reasons to think that it will generalise relatively easily to support non-syntactic 'semantic' concepts and the integration of syntax with semantics. The way in which the ICMAUS framework can support the parsing of NL provides a convenient way to introduce the concept of *multiple alignment* as it has been developed in this research.

Figure 1 shows how the French sentence 'e l l e s s o n t p e t i t e s' ("They are small", with feminine gender for "they") may be analysed into its constituent words and phrases by alignment with other patterns representing grammatical rules, stored in Old. The sentence, shown in row 0 of the alignment, is presented to SP61 as New.[1]

A pattern like 'S N #N VP #VP #S' in row 7 of the alignment corresponds to a re-write rule like 'S $\rightarrow$ N VP'. The grammatical patterns used in this and similar examples differ from conventional re-write rules because the re-write arrow is omitted, there is a 'termination' symbol at the end of the rule ('#S') and each of the 'code' symbols within the rule ('N' and 'VP' in this example) is paired with the corresponding termination symbol ('#N' and '#VP' respectively).

If we ignore rows 8 and 9 of Figure 1, the alignment is very much like the parsing that one would obtain using a context-free phrase-structure grammar (CF-PSG): it marks each level of the hierarchical structure of the sentence. What is different in this case is that the bottom two rows of the alignment mark 'discontinuous' dependencies of gender and number within the sentence: row 8 shows how the feminine subject ('e l l e s') is associated with the feminine form of the adjective ('p e t i t e s') at the end of the sentence; row 9 shows the association between the plural subject, plural verb and plural form of the following adjective.

This kind of ability to express discontinuous dependencies in syntax means that the system has more expressive 'power' than an unaugmented CF-PSG, possibly sufficient to express most aspects of NL structure in a succinct manner. Fuller discussion and examples may be found in (Wolff, 2000), including an example showing how the ICMAUS framework can accommodate the interesting pattern of inter-locking constraints in English auxiliary verbs.

[1] By convention, New is always shown in the top row of each alignment with patterns from Old underneath. The order of the rows below the first row is entirely arbitrary and has no special significance.

```
0             e l l e     s            s o n t              p e t i t     e s           0
              | | | |     |            | | | |              | | | | |     | |
1             | | | |     |            | | | |           A1 p e t i t #A1 | |           1
              | | | |     |            | | | |           |             |  | |
2             | | | |     |            | | | |    A PL F A1           #A1 e s #A        2
              | | | |     |            | | | |    | |  |                      |
3             | | | |     |       V PL s o n t #V | |  |                      |         3
              | | | |     |       | |          |  | |  |                      |
4             | | | |     |    VP V |          #V A |  |                      #A #VP    4
              | | | |     |    |  | |             | |  |                          |
5        N1 F e l l e #N1 |    |  | |             | |  |                          |     5
         |  |          |  |    |  | |             | |  |                          |
6   N PL N1 |         #N1 s #N |  | |             | |  |                          |     6
    | |     |               |  |  | |             | |  |                          |
7 S N |     |               #N VP | |             | |  |                         #VP #S 7
    | |     |                     | |             | |  |
8   N |     F                     V |             A |  F                                8
    | |                           | |             | |
9   N PL                          V PL            A PL                                  9
```

Figure 1: The best alignment found by SP61 with ‘e l l e s s o n t p e t i t e s’ in New and patterns representing grammatical rules in Old.

4.1 Production of Language

An interesting feature of the ICMAUS framework is that, *without any modification* it possible to use it to produce language as well as parse it. This is done by replacing the sentence in New with an encoded version of the sentence and running the model again. The result is an alignment which is, apart from the top row, the same as the alignment shown in Figure 1. Since it has the correct words in the correct order it is, in effect, an expression of the original sentence.

There is insufficient space here to describe more fully this aspect of the framework. It is similar in many ways to the way a suitably-designed Prolog program can be run forwards or backwards, depending on the data supplied. More explanation, with an example, can be found in (Wolff, 2000).

4.2 Semantics

Although no serious attempt has yet been made to apply the ICMAUS framework to NL processing with semantic structures, it is envisaged that this may be achieved by augmenting the grammatical patterns with additional symbols representing meanings.

5 Fuzzy Pattern Recognition and Best-Match Information Retrieval

NL parsing may be regarded as a kind of pattern recognition and the example just discussed may be regarded as an example of how the ICMAUS framework can support pattern recognition. However, what is missing from the example is the way the framework can achieve the kind of ‘fuzzy’ partial matching that is such a prominent feature of our abilities to recognise patterns and objects: in a typical scene, most objects are partially obscured by other objects but this does not normally prevent recognition. Within wide limits, any subset of the features of a pattern or object can be enough for it to be identified. As with NL processing, these abilities are singularly opaque to conscious introspection and are thus part of ‘non-conscious intelligence’.

Figure 2 shows a selection of alternative alignments found by SP61 between the mis-spelled word ‘i m t r n s i g a x n t’ (in New) and candidate words stored in Old. The alignments are in descending order of their compression score. In the manner of a spelling checker, SP61 identifies ‘i n t r a n s i g e n t’ as the most likely word. Notice how this is achieved despite errors of omission, commission and substitution.

Any example of ‘fuzzy’ pattern recognition like the one illustrated in Figure 2 may also be seen as an example of best-match information retrieval since the recognised pattern is, in effect, retrieved from memory. In the same way that we can recognise objects or patterns from a sub-set of their features, we have a remarkable ability to retrieve information from memory on the strength of fragmentary clues. As before, our ability is singularly free from conscious insights into how we manage to do what we can do.

5.1 Polythetic Classes

People not only have an ability to recognise things despite errors of omission, commission and substitution (as in the example just shown) but it seems that many of our concepts are *polythetic*, meaning that no single attribute

```
0     i m t r   n s i g a x n t     0
      |   | |   | | | |     | |
1 %I2 i n t r a n s i g e   n t #I2 1

0     i m t r n         s i g a x n t     0
      |       |         | | |     | |
1 %I2 i       n t r a n s i g e   n t #I2 1

0     i m t r n s       i g a x n t     0
      |       |         | |     | |
1 %I1 i       n t e l l i g e   n t #I1 1

0 i m t r n s i g a x n t     0
      |   |     |     | |
1 %T1 t a n     g e   n t #T1 1

0 i m t r n s i   g a x n t   0
          |       |
1 %U1 u   n t a n g l e s #U1 1
```

Figure 2: A selection of alignments found by SP61 between the mis-spelled word 'i m t r n s i g a x n t' in New and correctly spelled words stored in Old. The alignments are shown in descending order of their IC scores.

need appear in all members of the class. Although, in a naïf view of the nature of concepts, this aspect of 'natural' concepts may be puzzling, it can actually be modelled very simply by the use of re-write rules. For example, the following five rules: '1 → 2 3'; '2 → A'; '2 → B'; '3 → C'; and '3 → D', define the polythetic class {'AC', 'AD', 'BC', 'BD'}. Notice that none of the attributes 'A', 'B', 'C' or 'D' appear in all members of the class.

Given that these kinds of re-write rules can be modelled in the ICMAUS framework (as we saw in Section 4), it is clear that the framework can accommodate this aspect of human thinking.

5.2 Classes, Sub-Classes and Inheritance of Attributes

A prominent feature of the way we recognise objects and patterns is that we seem often to identify things at two or more levels of abstraction. For example, we may recognise something as a copy of *Pride and Prejudice*, which belongs in the class 'fiction', which itself belongs in the class 'book'.

This kind of multi-level recognition is illustrated in Figure 3. In the alignment, New (in row 0) contains a few of the features of the novel, while patterns from Old in the rows below represent different classes of entity, each one with characteristic attributes.

In effect, the alignment expresses the idea that the unknown entity described as 'Prd Aust cv pgs' has been recognised as an instance of *Pride and Prejudice*, that it is a work of fiction and that it may be classified as a book. From each of these classes, it *inherits* such attributes as being made of paper (from the class 'book'), that it is an inventive work (from the class 'fiction') and the particulars of its text (from the class *Pride and Prejudice*).

6 Probabilistic Reasoning

Another prominent feature of human non-conscious intelligence is our ability to make inferences about aspects of the world that we cannot immediately see or aspects of the world in the future. Such inferences are the fine-grained stuff of everyday thinking and, very often, we cannot easily articulate why it is that we think that a given situation is risky or that the weather may clear up in the afternoon. Such inferences are rarely certain and, generally, we have some subjective sense of the 'probability' that our expectations will be confirmed.

The example shown in Section 5.2 may be seen as an example of a similar kind of 'reasoning': from the features that are given ('Prd Aust cv pgs') we reason that the object we are dealing with is made of paper (from the class 'book'), that it is a work of inventive imagination (from the class 'fiction') and that the title contains the word 'Prejudice' (in addition to 'Pride') and that the author's first name is 'Jane' (from the class *Pride and Prejudice*).

In general, each column in any ICMAUS alignment that does **not** contain a symbol from New may be regarded as an inference or 'deduction' that may be drawn from the alignment.

In general, inferences that may be drawn from ICMAUS alignments are probabilistic. This is because, in general, there can be two or more alternative alignments for a given set of symbols in New. For example, if New contains the symbols 'warm blooded' and Old contains patterns for classes of animal, the system is likely to find one alignment that suggests that the unknown creature is a mammal and another alignment suggesting that it is a bird.

Each pattern in Old has an associated frequency of occurrence in some domain. From this frequency information, it is possible to calculate the absolute probability of each alignment and, for alternative alignments that match a given set of symbols in New, relative probabilities can be calculated too. The details of how the calculations can be done are described in (Wolff, 1999a).

6.1 Default Values and Nonmonotonic Reasoning

One of the difficulties in trying to use formal logic to model human-like everyday reasoning is that, in general, the former is 'monotonic'—meaning that the arrival of new information cannot invalidate deductions drawn earlier—while the latter is 'nonmonotonic'—meaning that inferences can be modified in the light of new information. For example, in traditional logic we may argue:

```
0            Prd               Aust     cv pgs                         0
              |                 |        |  |
1      bk tl  |      #tl au     |   #au cv pgs pap #bk                 1
       |  |   |       |  |      |    |              |
2    f bk |   |       |  |      |    |             #bk inv #f          2
     |    |   |       |  |      |    |                     |
3 pp f    tl Prd Prj #tl au Jn Aust #au                    #f text #pp 3
```

Figure 3: The best alignment found by SP61 with a few of the features of *Pride and Prejudice* in New and patterns representing relevant classes in Old. *Key:* au = author, Aust = Austin, bk = book, cv = cover, f = fiction, inv = invented story, Jn = Jane, pap = made of paper, pgs = divided into pages, pp = *Pride and Prejudice*, Prd = 'Pride', Prj = 'Prejudice', text = the text of the novel, tl = title.

"All birds fly; Tweety is a bird; therefore, Tweety can fly." This line of reasoning fails if we are told subsequently that Tweety is an ostrich (Ginsberg, 1994). Most formalisations of traditional logic cannot easily accommodate such statements as "Most birds fly" and yet that is the kind of generalisation we use all the time in our everyday thinking and which is needed to avoid the pitfalls of monotonic reasoning.

Figure 4 shows three alignments produced by SP61 with the pattern 'bird name Tweety #nm' in New and patterns representing generalisations about birds in Old. In row 1 of the first alignment, the pair of symbols 'canfly #cf' may be read as an empty variable for whether or not birds can fly. In effect, the pattern 'bird name #nm canfly #cf #bd' says that the value of this variable is undefined.

In row 2 of the second alignment, the symbol 'yes' appears between the symbols 'canfly #cf', saying in effect that birds can fly. The pattern 'bird canfly yes #cf #bd' expresses a *default value* ('yes') for the variable 'canfly #cf'. In effect, it says that if we know only that a creature is a bird, then we may guess that it probably can fly.

By contrast, row 2 of the third alignment says, in effect, that Tweety might be an ostrich and, as such, would not be able to fly.

The key point here is that the patterns 'bird name #nm canfly #cf #bd' and 'bird canfly yes #cf #bd' have a much higher frequency than the pattern 'ostrich bird canfly no #cf #bd #os'. This means that the relative probabilities of alignments (a) and (b) in the figure are much higher than the relative probability of (c). In effect, these three alignment say that, as a bird, it is probable that Tweety can fly but that there is a relatively low probability that Tweety might be an ostrich and, as such, he or she would not be able to fly. All of this accords with common sense.

What happens if we are told that Tweety is indeed an ostrich? We can represent this by modifying New to become 'ostrich name Tweety #nm'. In this case, there is only one alignment produced by SP61 that provides a match for all the symbols in New except 'Tweety'. This is shown in Figure 5.

Since there is now only one alignment that matches the critical symbols in New, the relative probability calculated by SP61 is 1.0. Since 'no' appears within the 'variable' 'canfly #cf', the system tells us, in effect, that, as an ostrich, it is certain that Tweety cannot fly. In accordance with the concept of nonmonotonic reasoning, the default inference that, as a bird, Tweety can fly has been overridden when we lean that Tweety is an ostrich.

6.2 Other Kinds of Human-Like Reasoning

It seems that the ICMAUS framework provides a flexible vehicle for a variety of other kinds of human-like reasoning (described in (Wolff, 1996, 1999a)):

- In addition to the use of patterns in Old to represent objects and classes of objects (as in the examples shown so far), patterns can be used to represent *if-then* associations like the association between leaf fall and the end of summer, between the sound of a gong and the serving of a meal, between turning a tap and the flow of water, and so on. Such patterns can be formed into alignments with corresponding inferences in much the same way as in the examples shown above.

- Chains of inferences ("if A then B, if B then C, ...") are easily modelled.

- The system can also model more subtle kinds of compound inference, that do not fit easily into the relatively rigid form dictated by systems like Prolog. An example is shown in (Wolff, 1999a).

- Apart from modelling a probabilistic kind of 'deductive' reasoning, the system can just as easily be used in 'reverse' to model abductive kinds of reasoning. Instead of "This thing can fly because it is an aeroplane", the system can form a set of alternative alignments that can be interpreted as "This thing might be an aeroplane or a bird or a bat because it can fly". As already indicated, absolute and relative probabilities can be calculated for alternative alignments of this kind.

- Given the translation of geometric patterns into the kind of one-dimensional patterns required by SP61, it is possible to model the kind of reasoning required for solving geometric analogy problems ("A is to B as C is to ?").

```
0 bird name Tweety #nm                0
    |    |           |
1 bird name         #nm canfly #cf #bd 1

(a)

0 bird name Tweety #nm                    0
    |    |           |
1 bird name         #nm canfly     #cf #bd 1
    |                     |         |    |
2 bird                  canfly yes #cf #bd 2

(b)

0        bird name Tweety #nm                        0
          |    |           |
1        bird name        #nm canfly    #cf #bd      1
          |                     |         |    |
2 ostrich bird                  canfly no #cf #bd #os 2

(c)
```

Figure 4: The three best alignments produced by SP61 with 'bird name Tweety #nm' in New and patterns representing generalisations about birds in Old.

```
0 ostrich      name Tweety #nm                       0
      |          |           |
1 ostrich bird   |           |  canfly no #cf #bd #os 1
            |    |           |    |         |    |
2         bird name         #nm canfly    #cf #bd      2
```

Figure 5: The best alignment produced by SP61 with 'ostrich name Tweety #nm' in New and patterns representing generalisations about birds in Old.

- In a manner not unlike our example of nonmonotonic reasoning (Section 6.1, above), the ICMAUS framework can model the phenomenon of 'explaining away': "If A implies B, C implies B, and B is true, then finding that C is true makes A *less* credible. In other words, finding a second explanation for an item of data makes the first explanation less credible." ((Pearl, 1988), his italics).

7 Unsupervised Inductive Learning

As we noted earlier, learning a first language is quite opaque to conscious insights—it is a good example of non-conscious intelligence. A working hypothesis in this programme of research is that insights gained from studying the way a child learns his or her first language are likely to be applicable to other kinds of learning too (more about this below).

As was noted earlier, this entire programme of research is based on earlier work on unsupervised inductive learning of language (Section 2), and the overall ICMAUS framework is designed to accommodate learning (Section 3). Most of the work to date has concentrated on areas other than learning but work is currently in progress developing SP70, a successor to SP61 designed as a full realisation of the ICMAUS framework, including unsupervised inductive learning. This section outlines current thinking about how the new model will work.

Imagine a new-born child listening to other people talking, either to the child directly or to each other. If ICMAUS principles apply, then each 'New' portion of speech will be encoded as far as possible in terms of what is already stored. In accordance with empiricist thinking about language learning, we shall suppose that, initially, the child has no knowledge of language patterns. Thus each New portion of speech cannot be encoded in terms of anything else and is simply stored 'as is'. After a time, however, as the child's brain gradually accumulates a collection of these speech patterns, new possibilities for economical encoding begin to appear.

Let us suppose that (the spoken equivalent of) 'i t s b e d t i m e n o w' is already stored. Then, at some point, the child may hear something like 'i t s p l a y t i m e n o w'. Looking for matching patterns, the child finds an alignment like the one shown at the top of Figure 6. By adding 'code' symbols at appropriate points, the child may convert the alignment into a fragment of 'grammar', something like what is shown in the bottom part of the figure.

```
Alignment:

0 i t s p l a y t i m e n o w 0
  | | |         | | | | | | |
1 i t s b e d   t i m e n o w 1

Derived fragment of 'grammar':

%1 i t s %2 #2 t i m e n o w #1
%2 0 p l a y #2
%2 1 b e d #2
```

Figure 6: An alignment and corresponding 'grammar' suggesting how a child may begin to identify words and classes of words in language.

Even at this early stage, the system has identified two discrete words and, in accordance with the principles of distributional linguistics, it has even recognised an embryonic grammatical class of 'qualifying' words: {'play', 'bed'}.

That distributional techniques can achieve this kind of thing is now widely recognised. Demonstrations include my MK10 model of the learning of speech segmentation (Wolff, 1975, 1977, 1980) and my SNPR model of syntax learning (Wolff, 1982, 1988).

7.1 Anticipated Features of SP70

This subsection indicates briefly some aspects of how the SP70 model is expected to work.

7.1.1 Sifting and Sorting

It is to be expected that many of the alignments found by SP70 will not be as 'tidy' as the example shown in Figure 6. It will not always happen that words or classes of words will be picked out as cleanly as in the example. How can the system learn 'clean' grammatical structures in the face of the inevitable messiness in the matching of linguistic patterns?

A related question is how children can learn 'correct' grammars despite the fact that the language that they hear is often corrupted in various ways. That learning systems of the kind we have been discussing can cope with 'dirty data' has been demonstrated already with the SNPR model of syntax learning.

The SNPR model can succeed in the face of dirty data because it is constantly searching for relatively large, relatively frequent patterns. Since 'errors' in the raw data are, by their nature, relatively rare, they are sifted out and discarded in favour of the 'good' patterns in the data.

As note earlier (Section 3), it is anticipated that, when it is fully developed, the ICMAUS framework will incorporate procedures for periodic purging of its knowledge structures, removing stored patterns that are not proving 'useful' in terms of the economical encoding of New information. It is anticipated that this kind of sifting and sorting of patterns will progressively remove the less satisfactory structures (resulting from untidy matching or dirty data) leaving a grammar largely composed of 'good' patterns.

7.1.2 Building Hierarchical Structures

The toy example shown in Figure 6 suggests how structures may be created at one level of abstraction above the raw data. It is anticipated that the same kinds of principles can be applied recursively so that patterns of code symbols formed in the early stages can themselves be incorporated into higher level structures. This kind of capability is incorporated in the SNPR model.

7.1.3 Generalization of Grammatical Rules and Correction of Overgeneralizations

Another important feature of the SNPR model is the way it can generalise grammatical rules and then correct overgeneralisations despite the fact that, by definition, all kinds of generalisations, correct and incorrect, have zero frequency in the corpus from which the grammar is induced. This is achieved in a totally 'unsupervised' way: without correction by a 'teacher', without the provision of 'negative' examples, and without any kind of 'grading' of the material from which the system learns.

MLE principles appear to provide the key to distinguishing between 'correct' and 'incorrect' generalisations without external error correction: 'correct' generalisations increase the compression that can be achieved while 'incorrect' generalisations do the opposite.

It is anticipated that similar principles and capabilities will be incorporated in SP70.

7.2 Unsupervised Inductive Learning of 'Semantic' Structures

The ICMAUS framework has been developed with the intention that it should be widely applicable, not confined narrowly to the syntax of natural language or some other circumscribed domain. One of the motivations for aiming for this kind of generality is that it should facilitate the learning of 'semantic' knowledge structures and the integration of syntax with semantics.

Figure 7 is intended to suggest how the kinds of class hierarchy considered in Section 5.2 may be learned. The alignment at the top of the figure is intended to suggest how the features that are shared by swans and robins may be identified. These shared features ('wings feathers beak') may be abstracted into a higher-level pattern describing the class 'bird' ('%bd wings feathers beak #bd') and then the two original patterns may be reduced to '%sw swan %bd #bd long-neck #sw' and '%rb robin %bd #bd red-breast #rb'.

```
Alignment:

0 swan  wings feathers beak long-neck  0
          |       |       |
1 robin wings feathers beak red-breast 1

Derived fragment of 'grammar':

%bd wings feathers beak #bd
%sw swan %bd #bd long-neck #sw
%rb robin %bd #bd red-breast #rb
```

Figure 7: An alignment and corresponding 'grammar' of non-linguistic patterns.

As with the learning of syntactic structures, it is anticipated that this kind of thing can be done recursively so that arbitrarily deep hierarchical structures may be built up. At some stage, there will be a need to integrate this kind of learning with the learning of syntactic structures to achieve the kind of knowledge that can serve the process of understanding language and the process of producing language from meanings.

8 Conclusion

I hope that this brief sketch of the possibilities offered by the ICMAUS framework will encourage other researchers to examine these ideas and consider whether or how they may be developed further.

Although the ICMAUS framework was not developed primarily as a model of non-conscious intelligence, it seems to offer useful insights into the abstract nature of several capabilities of humans (and animals) that are not generally accessible to conscious introspection.

References

Matthew L. Ginsberg. AI and nonmonotonic reasoning. In Dov. M. Gabbay, C. J. Hogger, and J. A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, pages 1–33. Oxford University Press, Oxford, 1994.

Ming Li and Paul Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer-Verlag, New York, 1997.

Judea Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, San Francisco, 1988.

J. G. Wolff. An algorithm for the segmentation of an artificial language analogue. *British Journal of Psychology*, 66:79–90, 1975. A copy may be obtained from www.sees.bangor.ac.uk/~gerry/lang_learn.html#wolff_1975.

J. G. Wolff. The discovery of segments in natural language. *British Journal of Psychology*, 68: 97–106, 1977. A copy may be obtained from www.sees.bangor.ac.uk/~gerry/lang_learn.html#wolff_1977.

J. G. Wolff. Language acquisition and the discovery of phrase structure. *Language Speech*, 23: 255–269, 1980. A copy may be obtained from www.sees.bangor.ac.uk/~gerry/lang_learn.html#wolff_1980.

J. G. Wolff. Language acquisition, data compression and generalization. *Language & Communication*, 2:57–89, 1982. A copy may be obtained from www.sees.bangor.ac.uk/~gerry/lang_learn.html#wolff_1982.

J. G. Wolff. Learning syntax and meanings through optimization and distributional analysis. In Yonata Levy, Izchak M. Schlesinger, and Martin D. S. Braine, editors, *Categories and Processes in Language Acquisition*, pages 179–215. Lawrence Erlbaum, Hillsdale, NJ, 1988. A copy may be obtained from www.sees.bangor.ac.uk/~gerry/lang_learn.html#wolff_1988.

J. G. Wolff. Learning and reasoning as information compression by multiple alignment, unification and search. In Alex Gammerman, editor, *Computational Learning and Probabilistic Reasoning*, pages 67–83. Wiley, Chichester, 1996. An earlier version was presented at Applied Decision Technologies '95, Brunel University, April 1995 (Proceedings of Stream 1, *Computational Learning and Probabilistic Reasoning*, 223-236, 1995).

J. G. Wolff. Probabilistic reasoning as information compression by multiple alignment, unification and search: an introduction and overview. *Journal of Universal Computer Science*, 5(7):418–462, 1999a. A copy may be obtained from www.jucs.org/wolff/jucs_5_7. The three articles on which this article is based may be obtained from www.iicm.edu/wolff/1998a, b, c.

J. G. Wolff. 'Computing' as information compression by multiple alignment, unification and search. *Journal of Universal Computer Science*, 5(11):777–815, 1999b. A copy may be obtained from www.jucs.org/wolff/jucs_5_11.

J. G. Wolff. Syntax, parsing and production of natural language in a framework of information compression by multiple alignment, unification and search. *Journal of Universal Computer Science*, 6(8):781–829, 2000. A copy may be obtained from www.jucs.org/wolff/jucs_6_8. Three articles that are the basis of this article may be obtained from www.iicm.edu/wolff/1998d1, d2, d3.

J. G. Wolff. Mathematics and logic as information compression by multiple alignment, unification and search. Technical report, School of Informatics, University of Wales at Bangor, 2001. A copy of the latest draft may be obtained from www.sees.bangor.ac.uk/∼gerry/sp_summary.html#maths_logic.