Information compression as a unifying principle in human learning, perception, and cognition

J Gerard Wolff[*]

December 7, 2018

Abstract

This paper reviews evidence for the idea that much of human learning, perception, and cognition, may be understood as information compression, and often more specifically as 'information compression via the matching and unification of patterns' (ICMUP). Evidence includes: information compression can mean selective advantage for any creature; the storage and utilisation of the relatively enormous quantities of sensory information would be made easier if the redundancy of incoming information were to be reduced; content words in natural languages, with their meanings, may be seen as ICMUP; other techniques for compression of information—such as class-inclusion hierarchies, schema-plus-correction, run-length coding, and part-whole hierarchies—may be seen in psychological phenomena; ICMUP may be seen in how we merge multiple views to make one, in recognition, in binocular vision, in how we can abstract object concepts via motion, in adaptation of sensory units in the eye of *Limulus*, the horseshoe crab, and in other examples of adaptation; the discovery of the segmental structure of language (words and phrases), grammatical inference, and the correction of over- and under-generalisations in learning, may be understood in terms of ICMUP; information compression may be seen in the perceptual *constancies*; there is indirect evidence for ICMUP in human cognition via kinds of redundancy such as the decimal expansion of π which are difficult for people to detect; much of the structure

[*] Dr Gerry Wolff, BA (Cantab), PhD (Wales), CEng, MBCS, MIEEE; CognitionResearch.org, Menai Bridge, UK; jgw@cognitionresearch.org; +44 (0) 1248 712962; +44 (0) 7746 290775; *Skype*: gerry.wolff; *Web*: www.cognitionresearch.org; ORCID iD 0000-0002-4624-8904.

and workings of mathematics—an aid to human thinking—may be understood in terms of ICMUP; and there is additional evidence via the *SP Theory of Intelligence* and its realisation in the *SP Computer Model*. Three objections to the main thesis of this paper are described, with suggested answers. These ideas may be seen to be part of a 'Big Picture' with six components, outlined in the paper.

The author declares that there is no conflict of interest regarding the publication of this paper.

1 Introduction

> "Fascinating idea! All that mental work I've done over the years, and what have I got to show for it? A goddamned zipfile! Well, why not, after all?" (John Winston Bush, 1996).

This paper describes empirical evidence for the idea that much of human learning, perception, and cognition, may be understood as information compression.[1] To be more specific, evidence will be presented that much of human learning, perception and cognition may be understood as information compression via the discovery of patterns that match each other, with the merging or 'unification' of two or more instances of any pattern to make one. References will also be made to the *SP Theory of Intelligence* and its realisation in the *SP Computer Model* in which information compression has a central role (Section 2.2.1).

Although this paper is primarily about information compression in human brains, it seems that similar principles apply throughout the nervous system, and throughout much of the animal kingdom. Accordingly, this paper has things to say here and there about the workings of neural tissue outside the human brain and in non-human species.

1.1 Abbreviations

For the sake of brevity in this paper: "information compression" may be shortened to 'IC'; the expression "information compression via the matching and unification of patterns" may be referred to as 'ICMUP'; and "human learning, perception, and cognition" may be 'HLPC'.

The main thesis of this paper—that much of HLPC may be understood as IC—may be referred to as 'ICHLPC'.

[1] This paper updates, revises, and extends the discussion in [104], itself the basis for [105, Chapter 2], but with the main focus on human learning, perception, and cognition.

For reasons given in Section 2.2, the name “SP” stands for *Simplicity* and *Power*.

The *SP Theory of Intelligence*, with its realisation in the *SP Computer Model*, may be referred to, together, as the *SP System*.

1.2 Presentation

In this paper: the next section (Section 2) describes some of the background to this research and some relevant general principles; the next-but-one section (Section 3) describes related research; Sections 4 to 20 inclusive describe relatively direct empirical evidence in support of ICHLPC; and Section 21 summarises indirect support for ICHLPC via the SP Theory of Intelligence;

Appendix A, referenced from Section 2.3 and elsewhere, gives some mathematical details relating to ICMUP and the SP System.

Appendix B, referenced from Section 3.1.1 and elsewhere, describes Horace Barlow’s change of view about the significance of IC in mammalian learning, perception, and cognition, with comments.

Appendix C, referenced from Section 22 and elsewhere, describes apparent contradictions of ideas in this paper, and how they may be resolved.

2 Background and general principles

This section provides some background to this paper and summarises some general principles that have a bearing on ICHLPC and the programme of research of which this paper is a part.

2.1 Seven variants of ‘information compression via the matching and unification of patterns’ (ICMUP)

This subsection fills out the concept of ICMUP, starting with the essentials, described in Section 2.1.1, next. Six variants of the basic idea are described in Sections 2.1.2 to 2.1.7.

Whilst care has been taken in this programme of research to avoid unnecessary duplication of information across different publications, the importance of the following seven variants of ICMUP has made it necessary, for the sake of clarity, to describe them quite fully both in this paper and also in [113].

2.1.1 Basic ICMUP

The main idea in ICMUP is illustrated in the top part of Figure 1. Here, a stream of raw data may be seen to contain two instances of the pattern '`INFORMATION`'. Subjectively, we 'see' this immediately. But in a computer or a brain, the discovery of that kind of replication of patterns must necessarily be done by some kind of searching for matches between patterns.

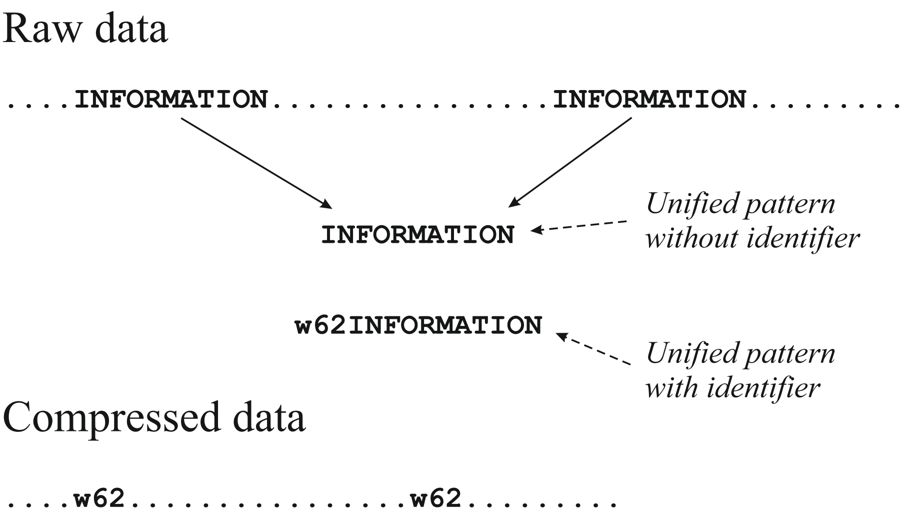

Raw data
....INFORMATION................INFORMATION.........
INFORMATION
Unified pattern without identifier
w62INFORMATION
Unified pattern with identifier
Compressed data
....w62................w62.........

Figure 1: A schematic representation of the way two instances of the pattern 'INFORMATION' in a body of raw data may be unified to form a single 'unified' pattern or 'chunk' of information, below the 'raw data'. Lower again in the figure, 'w62' is added to the unified chunk as a relatively short identifier or 'code'. The lowest part of the figure shows how the raw data may be compressed by replacing each instance of 'INFORMATION' with a copy of the short identifer. Adapted with permission from Figure 2.3 in [105].

In itself, the detection of repeated patterns is not very useful. But by merging or 'unifying' the two instances of '`INFORMATION`' in Figure 1 we may create the single instance shown below the raw data, thus achieving some compression of information in the raw data (Appendix A.1).

Other relevant points include:

- *Repetition of patterns and 'redundancy' in information.* From the perspective of ICMUP, the concept of *redundancy* in information may be seen as the occurrence of two or more arrays of symbols that match each other. As noted in Section 2.2.2, below, redundancy may take the form of good partial matches between patterns as well as exact matches between patterns.

- *A threshold on frequency of occurrence.* With regard to the previous point, an important qualification is that, for a given repeating array of symbols, **A**, to represent redundancy within a given body of information, **I**, **A**'s frequency of occurrence within **I** must be higher than would be expected by chance for an array of the same size [105, Sections 2.2.8.3 and 2.2.8.4].

- *Frequencies and sizes of patterns.* In connection with the preceding point: the minimum frequency needed to exceed the threshold is smaller for large patterns than it is for small patterns. Contrary to the common assumption that large frequencies are needed to attain statistical significance, frequencies as small as 2 can be statistically significant with patterns of quite moderate size or larger; and large patterns of a given frequency yield more compression than small ones of the same frequency (Appendix A.1, [105, Section 2.2.8.4]).

- *The concept of a 'chunk' of information.* A discrete pattern like '`INFORMATION`' is often referred to as a *chunk* of information, a term that gained prominence in psychology largely because of its use by George Miller in his influential paper *The magical number seven, plus or minus two* [60].

  Miller did not use terms like 'unification' or 'IC', and he sees some uncertainty in the significance of the concept of a chunk: "The contrast of the terms *bit* and *chunk* also serves to highlight the fact that we are not very definite about what constitutes a chunk of information." (p. 93, emphasis in the original). However, he describes how chunking of information may achieve something like compression of information: "... we must recognize the importance of grouping or organizing the input sequence into units or chunks. Since the memory span is a fixed number of chunks, we can increase the *number of bits of information that it contains* simply by building larger and larger chunks, each chunk containing more information than before." (p. 93, emphasis in the original) and "... the dits and dahs are organized by learning into patterns and ... *as these larger chunks emerge* the amount of message that the operator can remember increases correspondingly." (p. 93, emphasis in the original).

- *Basic ICMUP means lossy compression of information.* A point to notice about basic ICMUP of a body of information, **I**, is that, without the code mentioned above, it must always be 'lossy', meaning that non-redundant information in **I** will be lost. This is because, in the unification of two or more matching patterns in **I**, information is lost about the *location* of: 1) *all but one of those patterns* if the unified chunk is stored in one of the original locations within **I**; or alternatively 2) *all of those patterns* if the unified chunk is stored outside **I**.

2.1.2 Chunking-with-codes

The key idea with the *chunking-with-codes* variant of ICMUP is that each unified *chunk* of information (Section 2.1.1) receives a relatively short name, identifier, or *code*, and that code is used as a shorthand for the chunk of information wherever it occurs.

As already noted, this idea is illustrated in Figure 1, where, in the middle of the figure, the relatively short code or identifier '`w62`' is attached to a copy of the 'chunk' '`INFORMATION`', and we may suppose that that the pairing of code and unified chunk would be stored in some kind of 'dictionary', separate from the main body of data. Then, under the heading "Compressed data" at the bottom of the figure, each of the two original instances of '`INFORMATION`' is replaced by the short code '`w62`' yielding an overall compression of the original data.

Examples of chunking-with-codes from this paper are the use of 'ICMUP' as a shorthand for "information compression via the matching and unification of patterns", and 'HLPC' as a shorthand for "human learning, perception, and cognition".

The chunking-with-codes variant of ICMUP overcomes the weakness of basic ICMUP noted at the end of Section 2.1.1: that it loses non-redundant information about the *locations* of chunks in the original data, **I**. The problem may be remedied with chunking-with-codes because copies of the code for a given chunk may be used to mark the locations of each instance of the chunk within **I**.

Another point of interest is that, with the chunking-with-codes technique, compression of information may be optimised by assigning shorter codes to more frequent chunks and longer codes to rarer chunks, in accordance with some such scheme as Shannon-Fano-Elias coding [26, Section 5.9].

Similar principles may be applied in the other variants of ICMUP described in Sections 2.1.3 to 2.1.7, below.

2.1.3 Schema-plus-correction

The *schema-plus-correction* variant of ICMUP is like chunking-with-codes but the unified chunk of information may have variations or 'corrections' on different occasions.

An example from everyday life is a menu in a restaurant or café. This provides an overall framework, something like '`starter, main course, pudding`' which may be seen as a chunk of information. Each of the three elements of the menu may be seen as a place where each customer may make a choice or 'correction' to the menu. For example, one customer may choose '`starter(soup), main course(fish), pudding(apple pie)`' while another customer may choose '`starter(salad) main course(vegetable hotpot) pudding(ice cream)`', and so on.

The schema-plus-correction variant of ICMUP may achieve compression of information via two mechanisms:

- *The schema may itself have a short code.* In our menu example, each menu may have a short code such as '`bm`' for the breakfast menu, '`lm`' for the lunch-time menu, and so on.

- *Each 'correction' may have a short code.* Again with our menu example, options such as 'soup', 'fish', and so on, may each have a short code such as '`s`' for soup, '`f`' for fish, and so on.

With those two devices, a customer's order such as '`[lunch-time-menu: starter(soup), main course(fish), pudding(apple pie)]`' may be reduced to something like '`[lm: s, f, ap]`'.

2.1.4 Run-length coding

The *run-length coding* variant of ICMUP may be used with any sequence of two or more copies of a pattern where each copy except the first one follows immediately after the preceding copy. In that case, it is only necessary to record one copy of the pattern, with the number of copies, or with symbols or 'tags' to mark the start and end of the sequence.

For example, a repeated pattern like:

'`INFORMATIONINFORMATIONINFORMATIONINFORMATIONINFORMATION`'

may be reduced to something like '`INFORMATION(`×`5)`' (where '×5' records the number of instances of '`INFORMATION`'). Alternatively, the sequence may be reduced to something like '`p INFORMATION* #p`', where '`*`' means that "the pattern '`INFORMATION`' is repeated an unspecified number of times, and '`p ... #p`' specifies where the sequence begins and where it stops.

2.1.5 Class-inclusion hierarchy with inheritance of attributes

With the *class-inclusion hierarchy* variant of ICMUP, there is a hierarchy of classes and subclasses, with 'attributes' at each level. At every level except the top level, each subclass 'inherits' the attributes of all the higher levels.

For example, in simplified form, the class '`motorised vehicle`' contains sub-classes like '`road vehicle`' and '`rail vehicle`', the class '`road vehicle`' contains subclasses like '`bus`', '`lorry`', and '`car`', and so on. An attribute like '`contains engine`' would be assigned to the top level ('`vehicle`') and would be inherited by all lower-level classes, thus avoiding the need to record that information repeatedly at all levels in the hierarchy, and likewise for attributes at lower levels. Thus a class-inclusion hierarchy with inheritance of attributes combines IC with inference, in accordance with the close relation between those two things, noted in Section 2.5.

Of course there are many subtleties in the way people use class-inclusion hierarchies, such as cross-classification, 'polythetic' or 'family resemblance' concepts (in which no single attribute is necessarily present in every member of the given category and there need be no single attribute that is exclusive to that category [81]), and the ability to recognise that something belongs in a class despite errors of omission, commission, or substitution. The way in which the SP System can accommodate those kinds of subtleties is discussed in [105, Sections 2.3.2, 6.4.3, 12.2, and 13.4.6.2].

2.1.6 Part-whole hierarchy with inheritance of contexts

The *part-whole hierarchy* variant of ICMUP is like a class-inclusion hierarchy with inheritance of attributes except that the hierarchical structure represents the parts and subparts of some class or entity, and any given part inherits information about the context which it shares with all its siblings on the same level. A part-whole hierarchy promotes economy by sidestepping the need for each part of an entity at any given level to store full information about the higher-level structures of which it is a part—which is the same as other parts on the same level.

A simple example is the way that a '`person`' has parts like '`head`', '`body`', '`arms`', and '`legs`', while an arm may be divided into '`upper arm`', '`forearm`', '`hand`', and so on. In a structure like this, inheritance means that if one hears that a given person has an injury to his or her hand, one can infer immediately that that person's '`arm`' has been injured, and indeed his or her whole '`person`'.

2.1.7 SP-multiple-alignment as a generalised version of ICMUP

The seventh of the versions of ICMUP considered in this paper is the concept of SP-multiple-alignment, described in Section 2.2.2, below.

SP-multiple-alignment may be seen to be a generalised version of ICMUP which encompasses the other six versions described in Sections 2.1.1 to 2.1.6. How it can model those other six versions is described in detail in [111, Appendix B].

This versatility in modelling other versions of ICMUP is not altogether surprising since SP-multiple-alignment is largely responsible for the SP System's versatility in diverse aspects of intelligence (including diverse kinds of reasoning), in the representation of diverse kinds of knowledge, and its potential for the seamless integration of diverse aspects of intelligence and diverse kinds of knowledge, in any combination (Section 2.2.5).

2.2 The SP Theory of Intelligence

Readers will see that the paper contains references to the *SP Theory of Intelligence*, its realisation in the *SP Computer Model*, and associated ideas, especially the concept of *SP-multiple-alignment.* But it must be emphasised that the SP Theory is *not* the main focus of the paper. Instead it is relevant for subsidiary reasons:

- *Empirical evidence for ICHLPC strengthens empirical support for the SP Theory.* Since IC and, more specifically, ICMUP, are central in the SP Theory, empirical evidence for ICHLPC (presented in Sections 4 to 20) strengthens empirical support for the SP Theory, viewed as a theory of HLPC.

- *Direct empirical evidence for the SP Theory provides indirect evidence for ICHLPC.* Direct empirical evidence for the SP Theory—summarised in Section 2.2.5—provides indirect evidence for ICHLPC which is additional to that in in Sections 4 to 20 (see Section 21).

- *Clarifying theoretical issues related to HLPC.* The SP Computer Model, which may be seen as a working model of several aspects of HLPC, can help to clarify theoretical issues related to HLPC. It has, for example, proved useful in understanding issues discussed in Appendices B and C.

For those reasons, an outline of the theory is appropriate here.

2.2.1 Outline of the SP Theory of Intelligence: introduction

The *SP Theory of Intelligence* and its realisation in the *SP Computer Model*—the *SP System*—is a unique attempt to simplify and integrate observations and concepts across artificial intelligence, mainstream computing, mathematics, and human learning, perception, and cognition, with IC as a unifying theme. This broad scope for the SP programme of research has been adopted for reasons summarised in Section 2.6, below.

As mentioned in Section 1.1, the name "SP" stands for *Simplicity* and *Power*. This is because compression of any given body of information, **I**, may be seen as a process of reducing informational 'redundancy' in **I** and thus increasing its 'simplicity', whilst retaining as much as possible of its non-redundant expressive 'power'.

The SP Theory, the SP Computer Model, and some applications, are described quite fully in [107], and much more fully in [105]. Details of other publications about the SP System, most with download links, may be found on www.cognitionresearch.org/sp.htm. A download link for the source code of SP71, the latest version of the SP Computer Model, may be found under the heading 'SOURCE CODE' near the bottom of that page.

The SP Theory is conceived as a brain-like system as shown schematically in Figure 2. The system receives *New* information via its senses and stores some or all of it in compressed form as *Old* information.

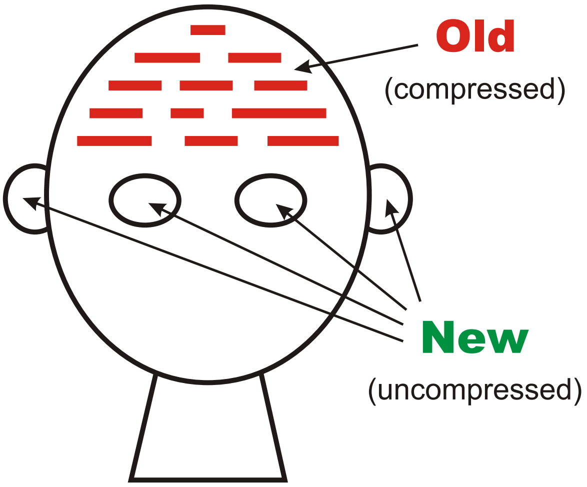


Figure 2: Schematic representation of the SP System from an 'input' perspective. Reproduced with permission from Figure 1 in [107].

All kinds of knowledge or information in the SP System are represented with arrays of atomic *SP-symbols* in one or two dimensions called *SP-patterns*. At present the SP Computer Model works only with one-dimensional SP-patterns but it is envisaged that, at some stage, it will be generalised to work with two-dimensional SP-patterns.

2.2.2 SP-multiple-alignment

A central part of the SP System is the powerful concept of *SP-multiple-alignment*, outlined here. The concept is described more fully in [107, Section 4] and [105, Sections 3.4 and 3.5].

The concept of SP-multiple-alignment in the SP System is derived from the concept of 'multiple sequence alignment' in bioinformatics (see, for example, [1]). That latter concept means an arrangement of two or more DNA sequences or sequences of amino-acid residues so that, by judicious 'stretching' of sequences in a computer, symbols that match from row to row are aligned—as illustrated in Figure 3. A 'good' multiple sequence alignment is one with a relatively high value for some metric related to the number of symbols that have been brought into line.

```
  G G A     G     C A G G G A G G A     T G     G   G G A
  | | |     |     | | | | | | | | |     | |     |   | | |
  G G | G   G C C C A G G G A G G A     | G G C G   G G A
  | | |     | | | | | | | | | | | |     | |     |   | | |
A | G A C T G C C C A G G G | G G | G C T G     G A | G A
  | | |           | | | | | | | | |   |   |     |   | | |
  G G A A         | A G G G A G G A   | A G     G   G G A
  | |   |         | | | | | | | |     |   |     |   | | |
  G G C A         C A G G G A G G     C   G     G   G G A
```

Figure 3: A 'good' multiple sequence alignment amongst five DNA sequences. Reproduced with permission from Figure 3.1 in [105].

For a given set of sequences, finding or creating 'good' multiple sequence alignments amongst the many possible 'bad' ones is normally a complex process—normally too complex to be solved by exhaustive search. For that reason, bioinformatics programs for finding good multiple sequence alignments use heuristic methods, building multiple sequence alignments in stages and discarding low-scoring multiple sequence alignments at the end of each stage, with backtracking or something equivalent to improve the robustness of the search.

With such methods it is not normally possible to guarantee that the best possible multiple sequence alignment has been found, but it is normally possible to find multiple sequence alignments that are good enough for practical purposes.

The two main differences between the concept of SP-multiple-alignment in the SP System and the concept of multiple sequence alignment in bioinformatics are that:

- *New and Old information.* With an SP-multiple-alignment, one of the SP-patterns (sometimes more than one) is *New* information from the system's environment (see Figure 2), and the remaining SP-patterns are *Old* information, meaning information that has been previously stored (also shown in Figure 2).

- *Encoding New information economically in terms of Old information.* In the creation of SP-multiple-alignments, the aim is to build ones that, in each case, allow the New SP-pattern (or SP-patterns) to be encoded economically in terms of the Old SP-patterns in the given SP-multiple-alignment. In each case, there is an implicit merging or unification of SP-patterns or parts of SP-patterns that match each other, as described in [107, Section 4.1] and [105, Section 3.5].

In the SP-multiple-alignment shown in Figure 4, one New SP-pattern is shown in row 0, and Old SP-patterns, drawn from a repository of Old SP-patterns, are shown in rows 1 to 9. By convention, the New SP-pattern(s) is always shown in row 0 and the Old SP-patterns are shown in the other rows, one SP-pattern per row.

```
0              f o r t u n e                      f a v o u r     s             t h e        b r a v e               0
               | | | | | | |                      | | | | | |     |             | | |        | | | | |
1              | | | | | | |                 Vr 6 f a v o u r #Vr |             | | |        | | | | |               1
               | | | | | | |                 |                 |  |             | | |        | | | | |
2              | | | | | | |             V 7 Vr               #Vr s #V          | | |        | | | | |               2
               | | | | | | |             |                          |           | | |        | | | | |
3              | | | | | | |        VP 3 V                          #V NP       | | |        | | | | |    #NP #VP    3
               | | | | | | |        |                                  |        | | |        | | | | |     |   |
4          N 4 f o r t u n e #N     |                                  |        | | |        | | | | |     |   |     4
           |                 |      |                                  |        | | |        | | | | |     |   |
5     NP 2 N                 #N #NP |                                  |        | | |        | | | | |     |   |     5
      |                          |  |                                  |        | | |        | | | | |     |   |
6 S 0 NP                        #NP VP                                 |        | | |        | | | | |     |  #VP #S 6
                                                                       |        | | |        | | | | |     |
7                                                                      |        | | |    N 5 b r a v e #N  |         7
                                                                       |        | | |    |             |   |
8                                                                      NP 1 D   | | | #D N             #N #NP        8
                                                                            |   | | | |
9                                                                           D 8 t h e #D                             9
```

Figure 4: The best SP-multiple-alignment produced by the SP Computer Model with a New SP-pattern representing a sentence to be parsed and a repository of user-supplied Old SP-patterns representing grammatical categories, including words. Reproduced with permission from Figure 2 in [109].

In this example, the New SP-pattern is a sentence and the Old SP-patterns in rows 1 to 9 represent grammatical structures including words. The overall effect of the SP-multiple-alignment is to 'parse' or analyse the sentence into its constituent parts and sub-parts, with each part marked with a category like '`NP`' (meaning 'noun phrase'), '`N`' (meaning 'noun'), '`VP`' (meaning 'verb phrase'), and so on. But, as described in Section 2.2.5, *the SP-multiple-alignment construct can do much more than parse sentences.*

Each SP-multiple-alignment is evaluated in terms of how it provides for the New SP-pattern in row 0 to being encoded economically in terms of the Old SP-patterns in the other rows. An SP-multiple-alignment is 'good' if the encoding is indeed economical. Details of how this is done are described in Appendix A.4.

With SP-multiple-alignments in the SP System, as with multiple sequence alignments in bioinformatics, the process of finding 'good' SP-multiple-alignments is too complex for exhaustive search, so it is normally necessary to use heuristic methods—which means that, as before, the best possible results may be missed but it is normally possible to find SP-multiple-alignments that are reasonably good.

At the heart of SP-multiple-alignment is a process for finding good full and partial matches between SP-patterns, described quite fully in [105, Appendix A]. As in the building of SP-multiple-alignments, heuristic search is an important part of the process of finding good full and partial matches between SP-patterns. Some details with relevant calculations are given in Appendix A.8.

As noted in Section 2.1.7, the concept of SP-multiple-alignment may be seen to be a generalised version of ICMUP, which encompasses all the other six variants of ICMUP described in Section 2.1.

2.2.3 Unsupervised learning in the SP System

Unsupervised learning in the SP System is described in [107, Section 5] and [105, Chapter 9]. In brief, it means searching for one or more collections of Old SP-patterns called *grammars* which are relatively good for the economical encoding of a given set of New SP-patterns.

As with the building of SP-multiple-alignments (Section 2.2.2), and the process of finding good full and partial matches between SP-patterns [105, Appendix A], and many other AI programs, unsupervised learning in the SP System uses heuristic techniques: doing the search in stages and, at each stage, concentrating the search in the most promising areas and cutting out the rest.

Some of the details of relevant calculations are given in Appendix A.7.

As mentioned in Section 2.2.4, learning in the SP System is quite different from the popular 'Hebbian' learning (often characterised as "Cells that fire together wire together"),[2] and it is quite different from how deep learning systems learn.

2.2.4 SP-Neural

Functionality that is similar to that of the SP System may be realised in a 'neural' sister to the SP System called *SP-Neural*, expressed in terms of neurons and their interconnections [109], as illustrated in Figure 5. Although the main elements of SP-Neural have been defined, there are details to be filled in. As with the development of the SP Theory itself, it is likely that many insights may be gained by building computer models of SP-Neural.

An important point here is that SP-Neural is quite different from the kinds of 'artificial neural network' that are popular in computer science, including those that provide the basis for 'deep learning' [76].

It is relevant to mention that Section V of [110] describes thirteen problems with deep learning in artificial neural networks and how, with the SP System, those problems may be overcome . The SP System also provides a comprehensive solution to a fourteenth problem with deep learning—"catastrophic forgetting"—meaning the way in which new learning in a deep learning system wipes out old memories [114].

Probably, SP-Neural's closest relative is Donald Hebb's [37] concept of a 'cell assembly' but, since learning in SP-Neural is likely to be modelled on learning in the SP System (Section 2.2.3), it will be quite different from Hebbian learning, and also quite different from learning in deep learning systems. More loosely, SP-Neural, when it is more fully developed, is likely to bear a superficial resemblance to Alan Turing's concept of an 'unorganised' machine [89] because its neural tissues would become progressively more organised as it learns.

2.2.5 Strengths and potential of the SP System

Largely because of the versatility of the SP-multiple-alignment construct, the SP System has strengths and potential in modelling several aspects of HLPC, as outlined here:

- *Versatility in aspects of intelligence.* The SP System has strengths in several aspects of human-like intelligence including: unsupervised

[2] Hebb's original version is "When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased." [37, p. 62].

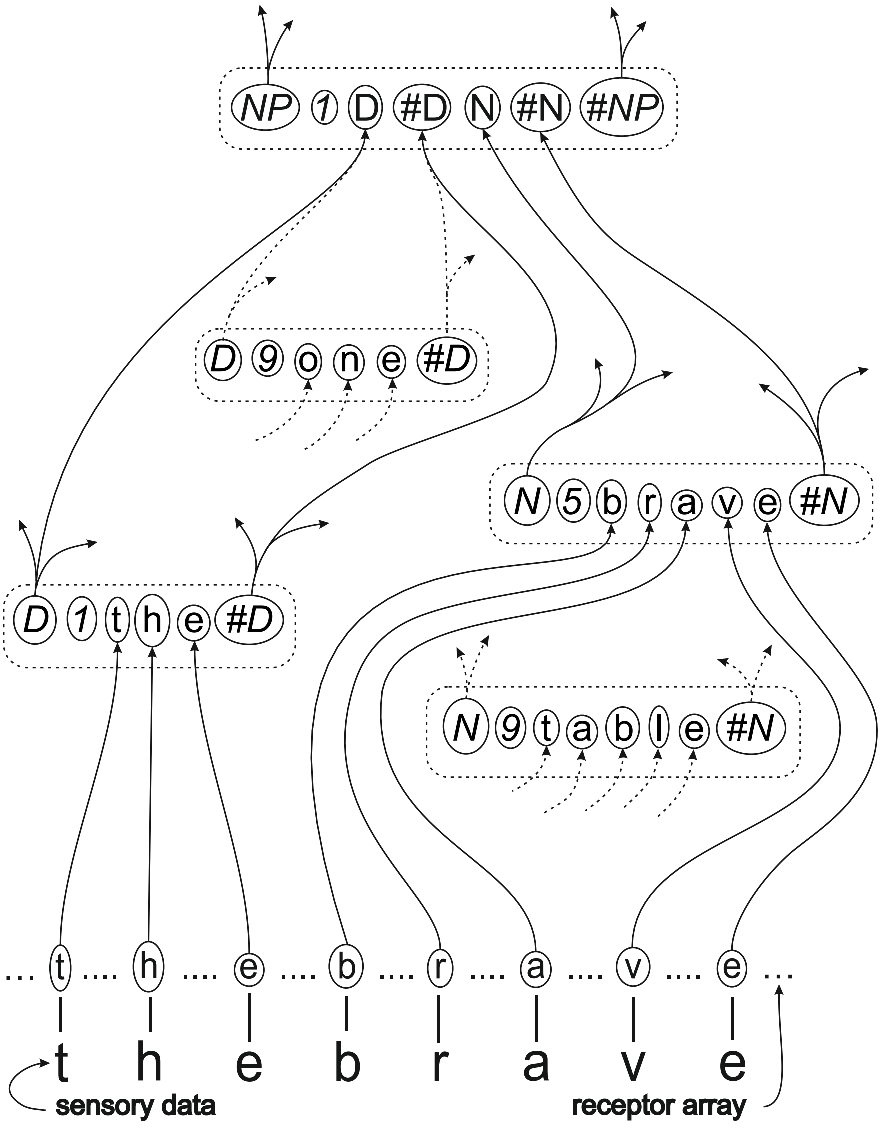


Figure 5: A schematic representation of a partial SP-multiple-alignment in SP-Neural, as discussed in [109, Section 4]. Each broken-line rectangle with rounded corners represents a *pattern assembly*—corresponding to an SP-pattern in the SP Theory. Each character or group of characters enclosed in a solid-line ellipse represents a *neural symbol* corresponding to an SP-symbol in the SP Theory. The lines between pattern assemblies represent nerve fibres with arrows showing the direction in which impulses travel. Neural symbols are mainly symbols from linguistics such as '`NP`' meaning 'noun phrase', '`D`' meaning a 'determiner', '`#D`' meaning the end of a determiner, '`#NP`' meaning the end of a noun phrase, and so on. Reproduced with permission from Figure 3 in [109].

learning, the analysis and production of natural language; pattern recognition that is robust in the face of errors in data; pattern recognition at multiple levels of abstraction; computer vision; best-match and semantic kinds of information retrieval; several kinds of reasoning (next bullet point); planning; and problem solving.

- *Versatility in reasoning.* Strengths of the SP System in reasoning include: one-step 'deductive' reasoning; chains of reasoning; abductive reasoning; reasoning with probabilistic networks and trees; reasoning with 'rules'; nonmonotonic reasoning and reasoning with default values; Bayesian reasoning with 'explaining away'[3]; causal reasoning; reasoning that is not supported by evidence; the already-mentioned inheritance of attributes in class hierarchies; and inheritance of contexts in part-whole hierarchies. There is also potential in the SP System for spatial reasoning and for what-if reasoning. Probabilities for inferences may be calculated in a straightforward manner (Appendix A.6).

- *Versatility in the representation and processing of knowledge.* The SP System has strengths in the representation and processing of several different kinds of knowledge including: the syntax of natural languages; class-inclusion hierarchies (with or without cross classification); part-whole hierarchies; discrimination networks and trees; if-then rules; entity-relationship structures; relational tuples; and concepts in mathematics, logic, and computing, such as 'function', 'variable', 'value', 'set', and 'type definition'. With the addition of Two-dimensional SP-patterns to the SP System, there is potential to represent such things as: photographs; diagrams; structures in three dimensions; and procedures that work in parallel.

- *Seamless integration of diverse aspects of intelligence and diverse kinds of knowledge, in any combination.* Because the SP System's versatility (in diverse aspects of intelligence and in the representation of diverse kinds of knowledge) flows from one relatively simple framework—SP-multiple-alignment—the system has clear potential for the seamless integration of diverse aspects of intelligence and diverse kinds of knowledge, in any combination. That kind of seamless integration appears to be *essential* in modelling the fluidity, versatility, and adaptability of the human mind.

[3] This means "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" [68, p. 7]. See also [107, Section 10.2] and [105, Section 7.8].

Figure 6 shows schematically how the SP System, with SP-multiple-alignment centre stage, exhibits versatility and integration.

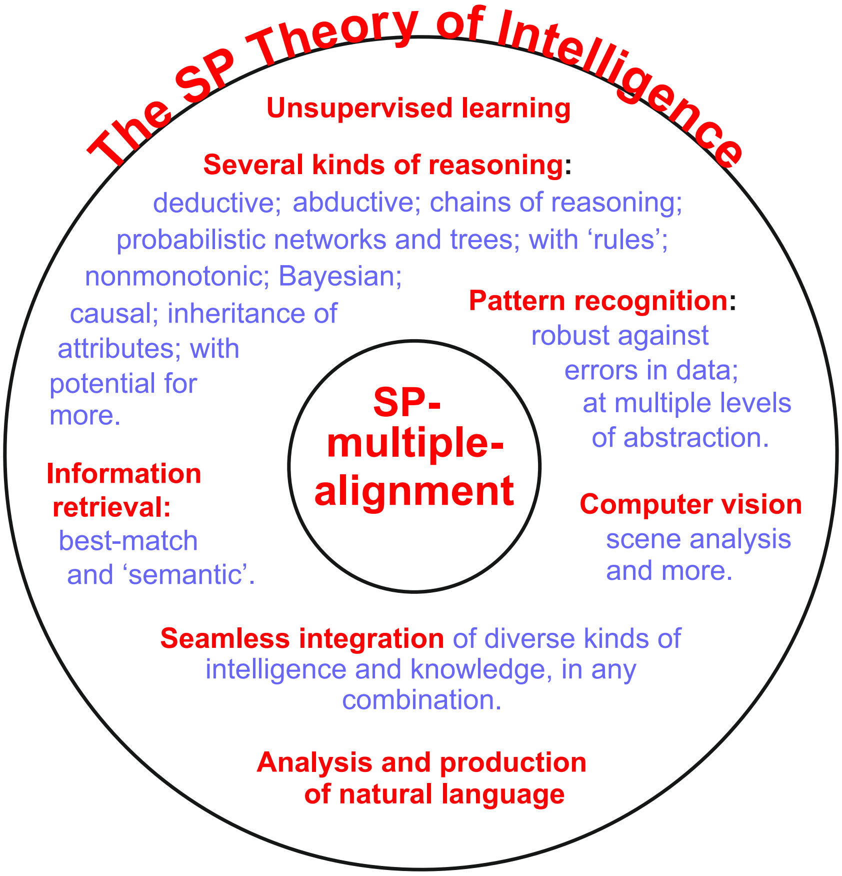

The SP Theory of Intelligence
Unsupervised learning
Several kinds of reasoning:
deductive; abductive; chains of reasoning; probabilistic networks and trees; with 'rules'; nonmonotonic; Bayesian; causal; inheritance of attributes; with potential for more.
Pattern recognition:
robust against errors in data; at multiple levels of abstraction.
SP-multiple-alignment
Information retrieval:
best-match and 'semantic'.
Computer vision
scene analysis and more.
Seamless integration of diverse kinds of intelligence and knowledge, in any combination.
Analysis and production of natural language

Figure 6: A schematic representation of versatility and integration in the SP System, with SP-multiple-alignment centre stage.

There is more detail in [112, Sections 4, 5, and 6], even more detail in [107], and most detail in [105]. Distinctive features and advantages of the SP System are described quite fully in [110].

How absolute and relative probabilities for SP-multiple-alignments may be calculated (for use in reasoning and other aspects of AI) is detailed in Appendix A.6.

2.2.6 Potential benefits and applications of the SP System

Apart from its strengths and potential in modelling aspects of the human mind, it appears that, in more humdrum terms, the SP System has several potential benefits and applications. These include: helping to solve nine problems with big data, helping to develop intelligence in autonomous robots, development of an intelligent database system, medical diagnosis, computer vision and natural vision, suggesting avenues for investigation in neuroscience, commonsense reasoning, and more. Details of relevant papers, with download links, may be found on www.cognitionresearch.org/sp.htm.

2.3 Avoiding too much dependence on mathematics

Many approaches to IC have a mathematical flavour (see, for example, [75]). Much the same is true of concepts of inference and probability which, as outlined in Section 2.5, are closely related to IC.

In the SP programme of research, the orientation is different. The SP Theory attempts to get below or behind the mathematics of other approaches to IC and to concepts of inference and probability: it attempts to focus on ICMUP, the relatively simple, 'primitive' idea that information may be compressed by finding two or more patterns that match each other, and merging or 'unifying' them so that multiple instances of the pattern are reduced to one.

That said, there is some mathematics associated with ICMUP, and there is some more which is incorporated in the SP Computer Model. They are described in Appendix A, and referenced at appropriate points throughout this paper.

There are four main reasons for this focus on ICMUP and the avoidance of too much dependence on mathematics:

- *Opening the door to non-mathematical mechanisms for compression of information.* Since ICMUP is relatively 'concrete' and less abstract than the more mathematical approaches to IC, it may open the door to non-mathematical mechanisms for compression of information which may otherwise be overlooked. Here are two putative examples:

  - *SP-multiple-alignment.* The concept of *SP-multiple-alignment* (Section 2.2.2) is founded on ICMUP and is not a recognised part of today's mathematics—but it has proved to be effective in the compression of information, it makes possible a relatively straightforward approach to the calculation of probabilities for inferences

(Appendix A.6), and it facilitates the modelling of several aspects of HLPC (Section 2.2.5, [105, 107]).

  - *ICMUP in SP-Neural.* Because SP-Neural (Section 2.2.4) is derived from the SP Theory, ICMUP is implicit in how, when it is more fully developed, SP-Neural is likely to work.

- *Don't use mathematics in describing the foundations of mathematics.* The SP Theory aims to be, amongst other things, a theory of the foundations of mathematics [113], so it would not be appropriate for the theory to be too dependent on mathematics.

- *The SP Theory is* **not** *founded on the concept of a universal Turing machine.* Whilst the SP Theory has benefitted from valuable insights gained from mathematically-oriented research on *Algorithmic Probability Theory*, *Algorithmic Information Theory*, and related work (Section 3.2), it differs from that work in that it is *not* founded on the concept of a 'universal Turing machine'.

  Instead, a focus on ICMUP, has yielded *a new theory of computing and cognition*, founded on ICMUP and SP-multiple-alignment, with the generality of the universal Turing machine [105, Chapter 4] but with strengths in the modelling of human-like intelligence which, as Alan Turing acknowledged [89, 98], are missing from the universal Turing machine (Section 2.2.5, [105, 107]).

- *ICMUP not obvious in such techniques as as wavelet compression and arithmetic coding.* At some abstract level, it may be that *all* mathematically-based techniques for compression of information are founded on ICMUP. And if the thesis of [113] is true, then all such techniques will indeed have an ICMUP foundation. But, nevertheless, techniques for the compression of information such as wavelet compression or arithmetic coding seem far removed from the simple idea of finding patterns that match each other and merging them into a single instance.

The SP System, including the concepts of SP-multiple-alignment and ICMUP, provides a novel approach to concepts of IC and probability (Section 2.5) which appears to have potential as an alternative to more widely recognised methods in these areas.

2.4 Empirical evidence and quantification

Although quantification of empirical evidence can in some studies be necessary or at least useful, it appears that, with most of the evidence presented in this paper (except in Sections 15 and 16), quantification would not be feasible or useful. In any case, attempts at quantification would be a distraction from the main thrust of the paper: that many examples of IC in HLPC are staring us in the face without the need for quantification.

As an example (from Section 6), a name like ‘New York’ is, in the manner of chunking-with-codes, a relatively brief ‘code’ for the enormously complex ‘chunk’ of information which is the structure and workings of the city itself. Similar things can be said about most other names for things, and also ‘content’ words like ‘house’, ‘table’ etc. In short, natural language may be seen to be a very powerful means of compressing information via the chunking-with-codes technique—and this is clear without the need for quantification.

2.5 IC and concepts of inference and probability

It has been recognised for some time that there is an intimate relation between IC and concepts of inference and probability (Appendix A.2, [78, 82, 83, 52]).

In case this seems obscure, it makes sense in terms of ICMUP. A pattern that repeats is one that invites ICMUP but it is also one that, via inductive reasoning, suggests possible inferences:

- *Any repeating pattern provides a basis for prediction.* Any repeating pattern—such as the association between black clouds and rain—provides a basis for prediction: black clouds suggest that rain may be on the way, and probabilities may be derived from the number of repetitions.

- *Inferences via partial matching.* With basic ICMUP and its variants, inferences may be made when one new pattern from the environment matches part of a stored, unified pattern. If, for example, we see ‘`INFORMA`’, we may guess, on the strength of the stored pattern, ‘`INFORMATION`’ (Figure 1), that the letters ‘`TION`’ are likely to follow. This idea is sometimes called ‘prediction by partial matching’ [86]. Of course, the pattern may be completed in a similar way if the incoming information is ‘`INFORMAN`’, ‘`INMATION`’, ‘`INFRMAION`’, and so on.

- *The SP System is designed to find partial matches as well as exact matches.* Because of the need to make inferences like those just described, and because a prominent feature of human perception is that we are rather good at finding good partial matches between patterns as well as exact matches, the SP System, including the process for building SP-multiple-alignments, is designed to search for redundancy in the form of good partial matches between patterns, as well as redundancy in the form of exact matches. This is done with a version of dynamic programming, described in [105, Appendix A].

There is a lot more detail about how this works with the SP-multiple-alignment concept in Appendix A.6, [107, Section 4.4] and [105, Section 3.7 and Chapter 7]. The SP System has proved to be an effective alternative to Bayesian theory in explaining such phenomena as ‘explaining away’ ([107, Section 10.2], [105, Section 7.8]).

As indicated in Section 4, the close connection between IC and concepts of inference and probability makes sense in terms of biology.

2.6 The Big Picture

The credibility of the ICHLPC thesis of this paper is strengthened by its position in a ‘Big Picture’ of the importance of IC in at least six areas:

- *Evidence for IC as a unifying principle in human learning, perception, and cognition.* This paper describes relatively direct empirical evidence for IC (and more specifically ICMUP) as a unifying principle in HLPC.

- *IC in the SP Theory of Intelligence.* ICMUP is central in the SP Theory of Intelligence (Section 2.2) which itself has much empirical and analytical support, summarised in Section 2.2.5, with pointers to where further information may be found.

- *IC in Neuroscience.* Because of its central role in the SP System, IC is central in *SP-Neural* (Section 2.2.4) and may thus have an important role in neuroscience.

- *IC and concepts of inference and probability.* It is known that there is an intimate relation between IC and concepts of inference and probability (Section 2.5).

- *IC as a foundation for mathematics.* The paper “Mathematics as information compression via the matching and unification of patterns” [113] argues that much of mathematics, perhaps all of it, may be understood in terms of ICMUP.

- *IC as a unifying principle in science.* It is widely agreed that “Science is, at root, just the search for compression in the world” [12, p. 247], with variations such as “Science may be regarded as the art of data compression” [52, p. 585], and more.

The Big Picture, as just outlined, is important for reasons summarised here:

- *You can’t play 20 questions with nature and win.* In his famous essay, “You can’t play 20 questions with nature and win”, Allen Newell [62] writes about the sterility of developing theories in narrow fields, and calls for each researcher to focus on “a genuine slab of human behaviour” (p. 303).[4]
- *Ockham’s razor.* Newell’s exhortation accords with a slightly extended version of Ockham’s razor: in developing simple theories of empirical phenomena, we should concentrate on those with the greatest explanatory range. Such theories will, naturally, be more useful than those with narrow scope, but, in addition, it seems that they are often relatively robust in the face of new evidence.
- *If you can’t solve a problem, enlarge it.* In a similar vein, President Eisenhower is reputed to have said: “If you can’t solve a problem, enlarge it”, meaning that putting a problem in a broader context may make it easier to solve. Good solutions to a problem may be hard to see when the problem is viewed through a keyhole, but become visible when the door is opened.

In keeping with these three reasons, the Big Picture is important in showing the potential of IC as a unifying principle across a wide canvass, including the six areas mentioned above.

Each of the six components of the Big Picture has support via empirical and analytical evidence which is specific to that component. In addition,

[4] Newell’s essay and his book *Unified Theories of Cognition* [63] led to many attempts by himself and others to develop such theories. But the difficulty of reaching agreement on a comprehensive framework for general, human-like AI is suggested by the following observation in [44, Locations 43–52]: “Despite all the current enthusiasm in AI, the technologies involved still represent no more than advanced versions of classic statistics and machine learning.” And what follows [44, Location 52] seems to confirm the persistence of the long-standing fragmentation of AI: “Behind the scenes, however, many breakthroughs are happening on multiple fronts: in unsupervised language and grammar learning, deep-learning, generative adversarial methods, vision systems, reinforcement learning, transfer learning, probabilistic programming, blockchain integration, causal networks, and many more”.

the six components are mutually supportive in the sense that the credibility of any one of them, including the main ICHLPC thesis of this paper, is strengthened via its position in the Big Picture.

Implications of the Big Picture include, for example, that IC should be a key part of any and all proposals for general, human-like AI, for theories of human learning, perception, and cognition, and for theories of cognitive neuroscience.

2.7 Volumes of data and speeds of learning

As noted in Section 2.1.1, large patterns may exceed the threshold for redundancy at a lower frequency than small patterns. With a complex pattern, such as an image of a person or a tree, there can be significant redundancy in a mere 2 occurrences of the pattern.

If redundancies can be detected via patterns that occur only 2 or 3 times in a given sample of data, unsupervised learning may prove to be effective with smallish amounts of data. This may help to explain why, in contrast to the very large amounts of data that are apparently required for success with deep learning, children and non-deep-learning types of learning program can do useful things with relatively tiny amounts of data [110, Section V-E].

In this connection, neuroscientist David Cox has been reported as saying: "To build a dog detector [with a deep learning system], you need to show the program thousands of things that are dogs and thousands that aren't dogs. My daughter only had to see one dog." and, the report says, she was happily pointing out puppies ever since.[5]

This issue relates to the way in which a camouflaged animal is likely to become visible when it moves relative to its background (Section 12). As with random-dot stereograms (Section 11), only two images which are similar but not the same are needed to reveal hidden structure.

2.8 Emotions and motivations

A point that deserves emphasis is that, while this paper is part of a programme of research aiming for simplification and integration of observations and ideas in HLPC and related fields, it does not aspire to be a comprehensive view of human psychology. In particular, it does not attempt to say anything about emotions or motivations, despite their undoubted importance and relevance to many aspects of human psychology, including cognitive psychology.

[5] "Inside the moonshot effort to finally figure out the brain", *MIT Technology Review*, 2017-10-12, bit.ly/2wRxsOg.

That said, it seems possible that IC might apply to emotions or motivations in the same way that it may be applied to sensory data and our concepts about the world.

3 Related research

An early example of thinking relating to IC in HLPC was the suggestion by William of Ockham in the 14th century that "Entities are not to be multiplied beyond necessity.". Later, Isaac Newton wrote that "Nature is pleased with simplicity" [64, p. 320], Albert Einstein wrote that "A theory is more impressive the greater the simplicity of its premises, the more different things it relates, and the more expanded its area of application.",[6] and more. Research with a more direct bearing on ICHLPC began in the 1950s and '60s after the publication of Claude Shannon's [78] 'theory of communication' (later called 'information theory'), and partly inspired by it.

In the two subsections that follow, there is a rough distinction between research with the main focus on issues in HLPC and neuroscience, and research that concentrates on issues in mathematics and computing. In both sections, research is described roughly in the order in which it was published.

In this research, the prevailing view of information, compression of information, and probabilities, is that they are things to be defined and analysed in mathematical terms. This perspective has yielded some useful insights but, as suggested in Section 2.3, there are potential advantages in the ICMUP perspective adopted in the SP research. This ICMUP perspective is what chiefly distinguishes the evidence which provides the main thrust of this paper from the related research described in this section.

3.1 Psychology-related and neuroscience-related research

Research relating to IC and HLPC and neuroscience may be divided roughly into two parts: early research initiated in the 1950s and '60s by Fred Attneave, Horace Barlow and others, and then after a relative lull in activity, later research from the 1990s onwards.

3.1.1 Early psychology-related and neuroscience-related research

In a paper called "Some informational aspects of visual perception", Fred Attneave [6] describes evidence that visual perception may be understood

[6] Quoted in [45, p. 512].

in terms of the distinction between areas in a visual image where there is much redundancy, and boundaries between those areas where non-redundant information is concentrated: "... information is concentrated along contours (i.e., regions where color changes abruptly), and is further concentrated at those points on a contour at which its direction changes most rapidly (i.e., at angles or peaks of curvature)." [6, p. 184].

For those reasons, he suggests that: "Common objects may be represented with great economy, and fairly striking fidelity, by copying the points at which their contours change direction maximally, and then connecting these points appropriately with a straight edge." [6, p. 185]. And he illustrates the point with a drawing of a sleeping cat reproduced in Figure 7.

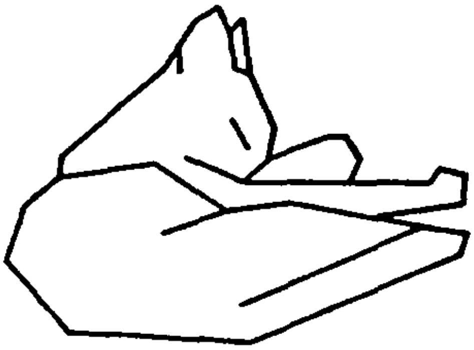


Figure 7: Drawing made by abstracting 38 points of maximum curvature from the contours of a sleeping cat, and connecting these points appropriately with a straight edge. Reproduced from Figure 3 in [6], with permission.

And he concludes with the suggestion that perception may be seen as economical description: "It appears likely that a major function of the perceptual machinery is to strip away some of the redundancy of stimulation, to describe or encode incoming information in a form more economical than that in which it impinges on the receptors." [6, p. 189].

Satosi Watanabe picked up the baton in a paper called "Information-theoretical aspects of inductive and deductive inference" [95]. He later wrote about the role of IC in pattern recognition [96, 97].

At about this time, Horace Barlow published a paper called "Sensory mechanisms, the reduction of redundancy, and intelligence" [8] in which he argued, on the strength of the large amounts of sensory information being fed into the [mammalian] central nervous system, that "the storage and utilization of this enormous sensory inflow would be made easier if the redundancy of the incoming messages was reduced." (p. 537). And he draws attention to evidence that, in mammals at least, each optic nerve is too small, by a wide margin, to carry reasonable amounts of the information impinging on the retina unless there is considerable compression of that information [8, p. 548].

In the paper, Barlow makes the interesting suggestion that "... the mechanism that organises [the large size of the sensory inflow] must play an important part in the production of intelligent behaviour." (p. 555), and in a later paper [9, p. 210] he writes:

> "... the operations required to find a less redundant code have a rather fascinating similarity to the task of answering an intelligence test, finding an appropriate scientific concept, or other exercises in the use of inductive reasoning. Thus, redundancy reduction may lead one towards understanding something about the organization of memory and intelligence, as well as pattern recognition and discrimination." .

These prescient insights into the significance of IC for the workings of human intelligence, with further discussion in [10], is a strand of thinking that has carried through into the SP Theory of Intelligence, with a wealth of supporting evidence, summarised in Section 2.2.5.[7]

Barlow developed these and related ideas over a period of years in several papers, some of which are referenced in this paper. However, in [11], he adopted a new position, arguing that:

> "... the [compression] idea was right in drawing attention to the importance of redundancy in sensory messages because this can often lead to crucially important knowledge of the environment, but it was wrong in emphasizing the main technical use for redundancy, which is compressive coding. The idea points to the enormous importance of estimating probabilities for almost everything the brain does, from determining what is redundant to fuelling Bayesian calculations of near optimal courses of action in a complicated world." (p. 242).

While there are some valid points in what Barlow says in support of his new position, his overall conclusions appear to be wrong. His main arguments are summarised in Appendix B, with what I'm sorry to say are my critical comments after each one.[8]

[7] When I was an undergraduate at Cambridge University, it was fascinating lectures by Horace Barlow about the significance of IC in the workings of brains and nervous systems, that first got me interested in those ideas.

[8] I feel apologetic about this because, as I mentioned, Barlow's lectures and his earlier research relating to IC in brains and nervous systems have been an inspiration for me over many years.

3.1.2 Later psychology-related and neuroscience-related research

Like the earlier studies, later studies relating to IC in brains and nervous systems have little to say about ICMUP. But they help to confirm the importance of IC in HLPC, and thus provide support for ICHLPC. A selection of publications are described briefly here.

Ruma Falk and Clifford Konold [29] describe the results of experiments indicating that the perceived randomness of a sequence is better predicted by various measures of its encoding difficulty than by its objective randomness. They suggest that judging the extent of a sequence's randomness is based on an attempt to encode it mentally, and that the subjective experience of randomness may result when that kind of attempt fails.

Jose Hernández-Orallo and Neus Minaya-Collado [40] propose a definition of intelligence in terms of IC. At the most abstract level, it chimes with remarks by Horace Barlow quoted in Section 3.1.1, and indeed it is consonant with the SP Theory itself. But the proposal shows no hint of how to model the kinds of capabilities that one would expect to see in any artificial system that aspires to human-like intelligence.

Nick Chater, with others, has conducted extensive research on HLPC, compression of information, and concepts of probability, generally with an orientation towards Algorithmic Information Theory, Bayesian theory, and related ideas. For example:

- Chater [18] discusses how 'simplicity' and 'likelihood' principles for perceptual organisation may be reconciled, with the conclusion that they are equivalent. He suggests that "the fundamental question is whether, or to what extent, perceptual organization is maximizing simplicity and maximizing likelihood." (p. 579).

- Chater [19] discusses the idea that the cognitive system imposes patterns on the world according to a simplicity principle, meaning that it chooses the pattern that provides the briefest representation of the available information. Here, the word 'pattern' means essentially a theory or system of one or more rules, a meaning which is quite different from the meaning of 'pattern' or 'SP-pattern' in the SP research, which simply means an array of atomic symbols in one or two dimensions. There is further discussion in [21].

- Emmanuel Pothos with Nick Chater [69] present experimental evidence in support of the idea that, in sorting novel items into categories, people prefer the categories that provide the simplest encoding of these items.

- Nick Chater with Paul Vitányi [22] describe how the 'simplicity principle' allows the learning of language from positive evidence alone, given quite weak assumptions, in contrast to results on language learnability in the limit [36]. There is further discussion in [41].

- Editors Nick Chater and Mike Oaksford [20] present a variety of studies using Bayesian analysis to understand probabilistic phenomena in HLPC.

- Paul Vitányi with Nick Chater [91] discuss whether it is possible to infer a probabilistic model of the world from a sample of data from the world and, via arguments relating to Algorithmic Information Theory, they reach positive conclusions.

Jacob Feldman [30] describes experimental evidence that, when people are asked to learn 'Boolean concepts', meaning categories defined by logical rules, the subjective difficulty of learning a concept is directly proportional to its 'compressibility', meaning the length of the shortest logically equivalent formula.

Don Donderi [27] presents a review of concepts that relate to the concept of 'visual complexity'. These include Gestalt psychology, Neural Circuit Theory, Algorithmic Information Theory, and Perceptual Learning Theory. The paper includes discussion of how these and related ideas may contribute to an understanding of human performance with visual displays.

Vivien Robinet and co-workers [73] describe a dynamic hierarchical chunking mechanism, similar to the MK10 Computer Model (Section 15). The theoretical orientation of this research is towards Algorithmic Information Theory, while the MK10 Computer Model embodies ICMUP.

From analysis and experimentation, Nicolas Gauvrit and others [34] conclude that how people perceive complexity in images seems to be partly shaped by the statistics of natural scenes. In [33], a slightly different grouping with Gauvrit as lead author describe how it is possible to overcome the apparent shortcoming of Algorithmic Information Theory in estimating the complexity of short strings of symbols, and they show how the method may be applied to examples from psychology.

In a review of research on the evolution of natural language, Simon Kirby and others [47] describe evidence that transmission of language from one person to another has the effect of developing structure in language, where 'structure' may be equated with compressibility. On the strength of further research, [85] conclude that increases in compressibility arise from learning processes (storing patterns in memory), whereas reproducing patterns leads to random variations in language.

On the strength of a theoretical framework, an experiment, and a simulation, Benoît Lemaire and co-workers [51] argue that the capacity of the human working memory may be better expressed as a quantity of information rather than a fixed number of chunks.

In related work, Fabien Mathy and Jacob Feldman [58] redefine George Miller's [60] concept of a 'chunk' in terms of Algorithmic Information Theory as a unit in a "maximally compressed code". On the strength of experimental evidence, they suggest that the true limit on short-term memory is about 3 or 4 distinct chunks, equivalent to about 7 uncompressed items (of average compressibility), consistent with George Miller's famous magical number.

And Mustapha Chekaf and co-workers [23] describe evidence that people can store more information in their immediate memory if it is 'compressible' (meaning that it conforms to a rule such as "all numbers between 2 and 6") than if it is not compressible. They draw the more general conclusion that immediate memory is the starting place for compressive recoding of information.

In addition to these several studies, there is quite a large body of research which relates to the concept of "efficient coding" in brains and nervous systems. These include the studies described in the following paragraphs.

Tiberiu Teşileanu, Bence Ölveczky, and Vijay Balasubramanian [87] developed a computer model of efficient two-stage learning, which proved accurate against data for the learning of birdsong by birds.

Ann Hermundstad and colleagues [39] found evidence in support of the propositions that efficient coding extends to higher-order sensory features, and that more neural resources are applied when sensory data is limited.

Vijay Balasubramanian [7] argues that the remarkable energy efficiency of the brain is achieved in part through the dedication of specialized circuit elements and architectures to specific computational tasks, in a hierarchy stretching from the scale of neurons to the scale of the entire brain, and that these structures are learned via an evolutionary process.

Francisco Heras and colleagues [38] provide evidence for mechanisms promoting energy efficiency in the workings of blowfly photoreceptors.

Biswa Sengupta and colleagues [77] investigate why the conversion of 'graded' potentials in the brain's neural circuits to 'action' potentials in those circuits is accompanied by substantial information loss and how this changes energy efficiency..

Simon Laughlin and Terrence Sejnowski [50] describe some of "the geometric, biophysical, and energy constraints that have governed the evolution of cortical networks", how "nature has optimized the structure and function of cortical networks with design principles similar to those used in electronic networks", and how "the brain ... exploits the adaptability of biological systems to reconfigure in response to changing needs."

Joseph Atick [4] reviews evidence relating to the principle that efficiency of information representation may be a design principle for sensory processing. In particular, it appears that this principle applies to large monopolar cells in the fly's visual system and retinal coding in mammals in the spatial, temporal and chromatic domains.

Joseph Atick and Norman Redlich [5] argue that the goal of processing in the retina is to transform the visual input as much as possible into a "statistically independent" form as a first step in creating a compressed representation in the cortex, as suggested by Horace Barlow. But the amount of compression that can be achieved in the retina is reduced by the need to suppress noise in the sensory input.

Adrienne Fairhall and colleagues [28] consider evidence relating to the optimisation of neural coding when the statistics of sensory data is changing. They conclude that "The speed with which information is optimized and ambiguities are resolved approaches the physical limit imposed by statistical sampling and noise."

Naama Brenner and colleagues [16] show that the input/output relation of a sensory system in a dynamic environment changes with the statistical properties of the environment. More specifically, when the dynamic range of inputs changes, the input/output relation rescales so as to match the dynamic range of responses to that of the inputs. And the scaling of the input/output relation is set to maximize information transmission for each distribution of signals.

William Bialek and colleagues [14] review progress on the question: "Does the brain construct an efficient representation of the sensory world?" In their answer to this question they take account of the biological value of sensory information, and they report preliminary evidence from studies of the fly's visual system which appear to support their view.

Stephanie Palmer and colleagues [67] show that efficient predictive computation starts at the earliest stages of the visual system, and that this is true of nearly every cell in the retina, and beyond. "Efficient representation of predictive information is a candidate principle that can be applied at each stage of neural computation."

Bruno Olshausen and David Field [66] discuss how "sparse coding" (the encoding of sensory information using a small number of active neurons at any given point in time) may confer several advantages and that there is evidence that "sparse coding could be a ubiquitous strategy employed in several different modalities across different organisms."

The same two authors, in [65], discuss the problem of how images can best be encoded and transmitted, with particular emphasis on how the eye and brain process visual information. They remark that "computer scientists and engineers now focusing on the problem of image compression should keep abreast of emerging results in neuroscience. At the same time, neuroscientists should pay close attention to current studies of image processing and image statistics."

Kristin Koch and colleagues [48] consider the question: how *much* information does the retina send to the brain and how is it apportioned among different cell types? They conclude that "With approximately 10^6 ganglion cells, the human retina would transmit data at roughly the rate of an Ethernet connection." This figure appears to be for the amount of information that is transmitted after decompression.

3.2 Mathematics-related and computer-related research

Other research, with an emphasis on issues in mathematics and computing, including artificial intelligence, can be helpful in the understanding of IC in brains and nervous systems. This includes:

- Ray Solomonoff developed Algorithmic Probability Theory showing the intimate relation between IC and inductive inference [82, 83] (Section 2.5).

- Chris Wallace with others explored the significance of IC in classification and related areas (see, for example, [93, 94, 3].

- Gregory Chaitin and Andrei Kolmogorov, working independently, developed Algorithmic Information Theory, building on the work of Ray Solomonoff. The main idea here is that the information content of a string of symbols is equivalent to the length of the shortest computer program that anyone has been able to devise that describes the string.

- Jorma Rissanen has developed related ideas in [71, 72] and other publications.

A detailed description of these and related bodies of research may be found in [52].

In research on deep learning in artificial neural networks, well reviewed by Jürgen Schmidhuber [76], there is some recognition of the importance of IC (in [76, Sections 4.2, 4.4, and 5.6.3]), but it appears that the idea is not well developed in deep learning systems.

Marcus Hutter, with others, [42, 43, 90] has developed the 'AIXI' model of intelligence based on Algorithmic Probability Theory and Sequential Decision Theory. He has also initiated the 'Hutter Prize', a competition with €50,000 of prize money, for lossless compression of a given sample of text. The competition is motivated by the idea that "being able to compress well is closely related to acting intelligently, thus reducing the slippery concept of intelligence to hard file size numbers."[9] This is an interesting project which may yet lead to general, human-level AI.

4 IC and biology

This section and those that follow (up to and including Section 21) describe evidence that, in varying degrees, lends support to the ICHLPC perspective. Most of this evidence comes directly from observations of people, but some of it comes from studies of animals—with the expectation that similar principles would be true of people.

First, let's take an abstract view of why IC might be important in people and other animals. In terms of biology, IC can confer a selective advantage to any creature by allowing it to store more information in a given storage space or use less storage space for a given amount of information, and by speeding up the transmission of any given volume of information along nerve fibres—thus speeding up reactions—or reducing the bandwidth needed for the transmission of the same volume of information in a given time.

Perhaps more important than the impact of IC on the storage or transmission of information is the close connection, outlined in Section 2.5, between IC and concepts of inference and probability. Compression of information provides a means of predicting the future from the past and estimating probabilities so that, for example, an animal may learn to predict where food may be found or where there may be dangers.

As mentioned in Section 2.5, the close connection between IC and concepts of inference and probability makes sense in terms of ICMUP: any repeating pattern can be a basis for inferences, and the probabilities of such inferences may be derived from the number of repetitions of the given pattern.

Being able to make inferences and estimate probabilities can mean large savings in the use of energy and other benefits in terms of survival.

[9] From www.hutter1.net, retrieved 2017-10-10.

5 Sensory inflow, redundancy, and the transmission and storage of information

As mentioned in Section 3.1.1, Fred Attneave [6] describes how visual perception may be understood in terms of the distinction between areas in a visual image where there is much redundancy and boundaries between those areas where non-redundant information is concentrated. And he suggests that visual perception may be understood, at least in part, as the economical description of sensory input.

Also mentioned in the same section is Horace Barlow's [8] argument that compression of sensory information is needed to cope with the large volumes of such information, and, more specifically, his recognition that, without compression of the information falling on the retina, each optic nerve would be too small to transmit reasonable amounts of that information to the brain [8, p. 548].

6 Chunking-with-codes

ICMUP is so much embedded in our thinking, and seems so natural and obvious, that it is easily overlooked. This section, with Sections 7 and 8, describe some examples.

In the same way that 'TFEU' may be a convenient code or shorthand for the rather cumbersome expression 'Treaty on the Functioning of the European Union' (Appendix C.1.2), a name like 'New York' is, as previously noted in Section 2.4, a compact way of referring to the many things and activities in that renowned city . Likewise for the many other names that we use: 'Nelson Mandela', 'George Washington', 'Mount Everest', and so on.

The 'chunking-with-codes' variant of ICMUP (Section 2.1.2) permeates our use of natural language, both in its surface forms and in the way in which surface forms relate to meanings.[10]

Because of its prominence in natural language and because of its intrinsic power, chunking-with-codes is probably important in non-verbal aspects of our thinking, as may be inferred from empirical support for the SP System and its strengths in several aspects of intelligence (Section 2.2.5).[11]

[10] Although natural language provides a very effective means of compressing information about the world, it is not free of redundancy. And that redundancy has a useful role to play in, for example, enabling us to understand speech in noisy conditions, and in learning the structure of language. How this apparent inconsistency may be resolved is discussed in Appendix C.2.

[11] Contrary to the view which is sometimes expressed that thinking is not possible with-

Ever since George Miller's influential paper [60], the concept of a 'chunk' has been the subject of much research in psychology and related disciplines (see, for example, [2, 115, 74, 35]).

Principles outlined in this section are likely to apply also to variants of ICMUP discussed in Sections 7 and 8, below.

7 Class-inclusion hierarchies

As with chunking-with-codes, class-inclusion hierarchies, with variations such as cross-classification, are prominent in our use of language and in our thinking. Benefits arise from economies in the storage of information and in inferences via inheritance of attributes, in accordance with the 'class-inclusion hierarchies' variant of ICMUP (Section 2.1.5).

As with chunking-with-codes, names for classes of things provide for great economies in our use of language: most 'content' words (nouns, verbs, adjectives, and adverbs) in our everyday language stand for *classes* of things and, as such, are powerful aids to economical description.

Imagine how cumbersome things would be if, on each occasion that we wanted to refer to a "table", we had to say something like "A horizontal platform, often made of wood, used as a support for things like food, normally with four legs but sometimes three, ...", like the slow *Entish* language of the Ents in Tolkien's *The Lord of the Rings*.[12] Similar things may be said for verbs like "speak" or "dance", adjectives like "artistic" or "exuberant", and adverbs like "quickly" or "carefully".

Classes and categories have been the subject of much research in psychology and related disciplines over several decades (see, for example, [49, 53, 59]).

8 Schema-plus-correction, run-length coding, and part-whole hierarchies

As with chunking-with-codes and class-inclusion hierarchies, it seems natural to conceptualise things in terms of other techniques described in Section

out language, there is evidence in [32] for non-verbal thinking by congenitally deaf people without knowledge of written or spoken natural language, and there is other evidence in [13] for non-verbal thinking in people and in animals.

[12] J. R. R. Tolkien, *The Lord of the Rings*, London: HarperCollins, 2005, Kindle edition. For a description of Entish, see, for example, page 480. See also, pages 465, 468, 473, 477, 478, 486, and 565.

2.1. In all cases, there is clear potential for substantial economies in how knowledge is represented and for the making of useful inferences.

8.1 Schema-plus-correction

As mentioned in Section 2.1.3, a menu in a restaurant or café is an obvious example of the schema-plus-correction device in everyday thinking. Other examples are the uses of forms to gather information about candidates for a job, the features of a house for sale, a check-list for repairs on a car, and so on. And knowledge of almost any skill such as baking a cake, gardening, or woodwork, may be seen as a schema that may be tailored for a specific task—such as baking a coffee-and-walnut cake—by plugging in values for that task.

An interesting example of schema-plus-correction in everyday life is the UK shipping forecast which leaves out most of the schema and gives only the corrections to the schema. So, for example: "good, becoming moderate or poor" refers to visibility without mentioning that word; "moderate or rough" refers to the state of the sea, without mentioning that expression; figures for wind speed are given without mentioning that they refer to the Beaufort wind force scale; a word like "later" means a time that is more than 12 hours from the time the forecast was issued; and so on.

8.2 Run-length coding

If anything is repeated, especially if it repeated a large number of times, it seems natural and obvious to describe the repetition with a form of run-length coding. For example, an instruction to walk from one place to another may be: "From the old oak tree keep walking until you see the river". Here, "the old oak tree" marks the start of the repetition, "keep walking" describes the repeated operation of putting one foot in front of the other, and "until you see the river" marks the end of the repetition.

8.3 Part-whole hierarchies

As with class-inclusion hierarchies, part-whole hierarchies are prominent in our language and in our thinking. In describing anything that is more complex than 'very simple', such as a house or a car, it seems natural and obvious to divide it into parts and sub-parts through as many levels as are needed, thus promoting economies and the making of inferences as described in Section 2.1.6.

9 Merging multiple views to make one

Here is another example of something that is so familiar that we are normally not aware that it is part of our perceptions and thinking.

If, when we are looking at something, we close our eyes for a moment and open them again, what do we see? Normally, it is the same as what we saw before. But creating a single view out of the before and after views, means unifying the two patterns to make one and thus compressing the information, as shown schematically in Figure 8.[13]

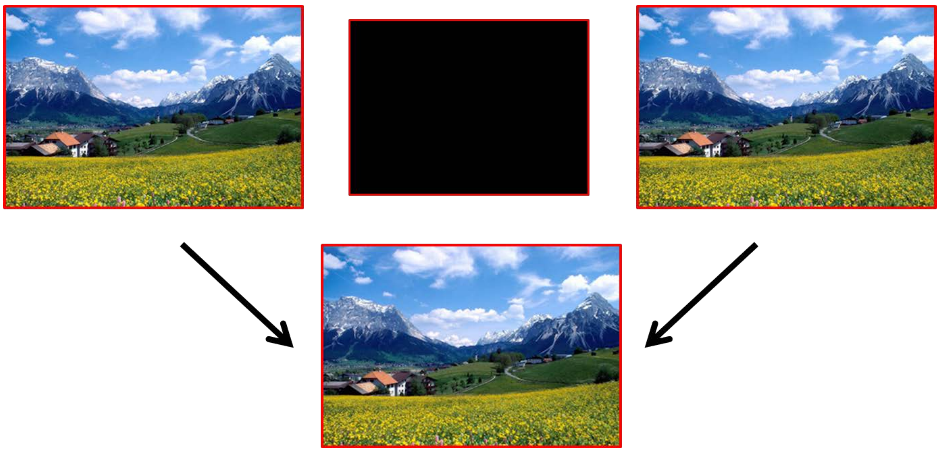


Figure 8: A schematic view of how, if we close our eyes for a moment and open them again, we normally merge the before and after views to make one. The landscape here and in Figure 9 is from Wallpapers Buzz (www.wallpapersbuzz.com), reproduced with permission.

It seems so simple and obvious that if we are looking at a landscape like the one in the figure, there is just one landscape even though we may look at it two, three, or more times. But if we did not unify successive views we would be like an old-style cine camera that simply records a sequence of frames, without any kind of analysis or understanding that, very often, successive frames are identical or nearly so.

[13] It is true that people may, on occasion, not detect large changes to objects and scenes ('change blindness') [80] and that, without attention, we may not even perceive objects ('inattentional blindness') [79], but it is also true that we can detect differences between pairs of images that are similar but not identical—which means that we can also detect the similarities between such pairs of images. That ability to detect similarities, together with our ordinary experience that we normally merge multiple views to make one, as described in the main text, implies that compression of information is an important part of visual perception.

10 Recognition

With the kind of merging of views just described, we do not bother to give it a name. But if the interval between one view and the next is hours, months, or years, it seems appropriate to call it 'recognition'. In cases like that, it is more obvious that we are relying on memory, as shown schematically in Figure 9. Notwithstanding the undoubted complexities and subtleties in how we recognise things, the process may be seen in broad terms as ICMUP: matching incoming information with stored knowledge, merging or unifying patterns that are the same, and thus compressing the information.

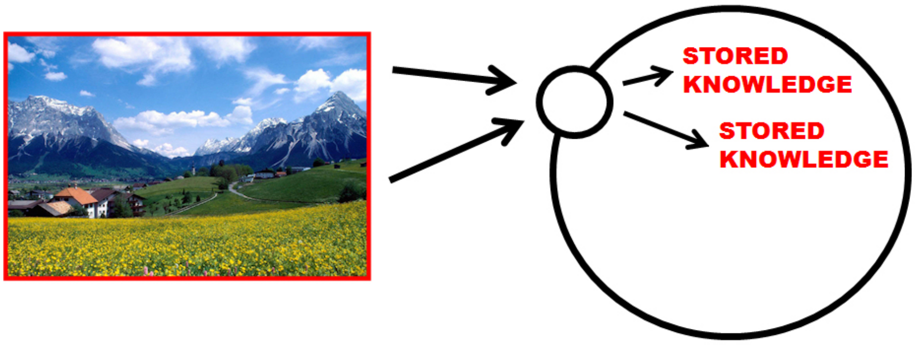


Figure 9: Schematic representation of how, in recognition, incoming visual information may be matched and unified with stored knowledge.

If we did not compress information in that way, our brains would quickly become cluttered with millions of copies of things that we see around us—people, furniture, cups, trees, and so on—and likewise for sounds and other sensory inputs.

As mentioned earlier, Satosi Watanabe has explored the relationship between pattern recognition and IC [96, 97].

11 Binocular vision

ICMUP may also be seen at work in binocular vision:

> "In an animal in which the visual fields of the two eyes overlap extensively, as in the cat, monkey, and man, one obvious type of redundancy in the messages reaching the brain is the very nearly exact reduplication of one eye's message by the other eye." [9, p. 213].

In viewing a scene with two eyes, we normally see one view and not two. This suggests that there is a matching and unification of patterns, with a corresponding compression of information. A sceptic might say, somewhat implausibly, that the one view that we see comes from only one eye. But that sceptical view is undermined by the fact that, normally, the one view gives us a vivid impression of depth that comes from merging the two slightly different views from both eyes.

Strong evidence that, in stereoscopic vision, we do indeed merge the views from both eyes, comes from a demonstration with 'random-dot stereograms', as described in [108, Section 5.1] (see also Appendix A.3).

In brief, each of the two images shown in Figure 10 is a random array of black and white pixels, with no discernable structure, but they are related to each other as shown in Figure 11: both images are the same except that a square area near the middle of the left image is further to the left in the right image.

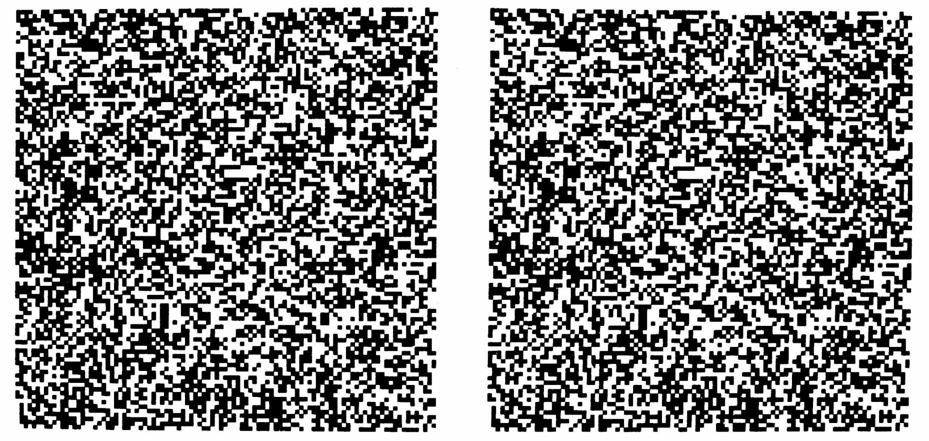
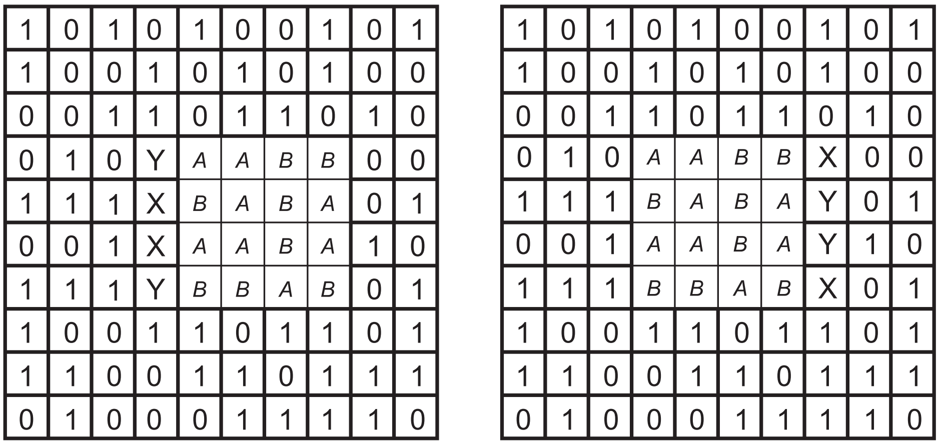


Figure 10: A random-dot stereogram from [46, Figure 2.4-1], reproduced with permission of Alcatel-Lucent/Bell Labs.

When the images in Figure 10 are viewed with a stereoscope, projecting the left image to the left eye and the right image to the right eye, the central square appears gradually as a discrete object suspended above the background.

Although this illustrates depth perception in stereoscopic vision—a subject of some interest in its own right—the main interest here is on how we see the central square as a discrete object. There is no such object in either of the two images individually. It exists purely in the *relationship* between the two images, and seeing it means matching one image with the other and



1	0	1	0	1	0	0	1	0	1
1	0	0	1	0	1	0	1	0	0
0	0	1	1	0	1	1	0	1	0
0	1	0	Y	A	A	B	B	0	0
1	1	1	X	B	A	B	A	0	1
0	0	1	X	A	A	B	A	1	0
1	1	1	Y	B	B	A	B	0	1
1	0	0	1	1	0	1	1	0	1
1	1	0	0	1	1	0	1	1	1
0	1	0	0	0	1	1	1	1	0

1	0	1	0	1	0	0	1	0	1
1	0	0	1	0	1	0	1	0	0
0	0	1	1	0	1	1	0	1	0
0	1	0	A	A	B	B	X	0	0
1	1	1	B	A	B	A	Y	0	1
0	0	1	A	A	B	A	Y	1	0
1	1	1	B	B	A	B	X	0	1
1	0	0	1	1	0	1	1	0	1
1	1	0	0	1	1	0	1	1	1
0	1	0	0	0	1	1	1	1	0

Figure 11: Diagram to show the relationship between the left and right images in Figure 10. Reproduced from [46, Figure 2.4-3], with permission of Alcatel-Lucent/Bell Labs.

unifying the parts which are the same.

This example shows that, although the matching and unification of patterns is a usefully simple idea, there are interesting subtleties and complexities that arise in finding a good match when the two patterns are similar but not identical.

11.1 Finding a good match

Seeing the central object in a random-dot stereogram means finding a good match between relevant pixels in the central area of the left and right images, and likewise for the background. Here, a good match is one that yields a relatively high level of IC. Since there is normally an astronomically large number of alternative ways in which combinations of pixels in one image may be aligned with combinations of pixels in the other image, it is not normally feasible to search through all the possibilities exhaustively.

11.2 The best is the enemy of the good

As with the SP System (Sections 2.2.1 to 2.2.3) and many problems in artificial intelligence, the best is the enemy of the good. Instead of looking for the perfect solution—which may lead to outright failure—we can do better, achieving something useful on most occasions by looking for solutions that are good enough for practical purposes. With this kind of problem, acceptably good solutions can often be found in a reasonable time with heuristic search. One such method for the analysis of random-dot stereograms has been described by Marr and Poggio [56].

12 Abstracting object concepts via motion

It seems likely that the kinds of processes that enable us to see a hidden object in a random-dot stereogram also apply to how we see discrete objects in the world. The contrast between the relatively stable configuration of features in an object such as a car, compared with the variety of its surroundings as it travels around, seems to be an important part of what leads us to conceptualise the object as an object [108, Section 5.2].

Any creature that depends on camouflage for protection—by blending with its background—must normally stay still. As soon as it moves relative to its surroundings, it is likely to stand out as a discrete object ([108, Section 5.2], see also Section 2.7).

The idea that IC may provide a means of discovering 'natural' structures in the world—such as the many objects in our visual world—has been dubbed the 'DONSVIC' principle: *the discovery of natural structures via information compression* [107, Section 5.2]. Of course, the word 'natural' is not precise, but it has enough precision to be a meaningful name for the process of learning the kinds of concepts which are the bread-and-butter of our everyday thinking.

Similar principles may account for how young children come to understand that their first language (or languages) is composed of words (Section 15).

13 Adaptation in the eye of *Limulus* and run-length coding

IC may also be seen down in the works of vision. Figure 12 shows a recording from a single sensory cell (*ommatidium*) in the eye of a horseshoe crab (*Limulus polyphemus*), first when the background illumination is low, then when a light is switched on and kept on for a while, and later switched off—shown by the step function at the bottom of the figure.

Perhaps contrary to what one might expect—a low rate of firing when illumination is low—the ommatidium fires at a moderate 'background' rate of about 20 impulses per second when the illumination is low (shown at the left of the figure). When the light is switched on, the rate of firing increases sharply but instead of staying high while the light is on (as one might expect),

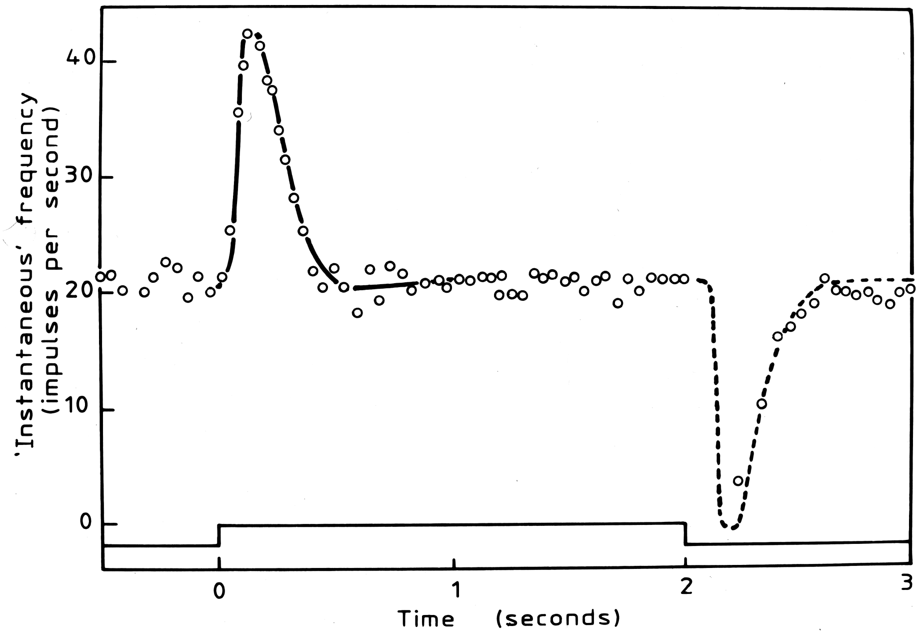


Figure 12: Variation in the rate of firing of a single ommatidium of the eye of a horseshoe crab in response to changing levels of illumination. Reproduced from [70, Figure 16], with permission from the Optical Society of America.

it drops back almost immediately to the background rate. The rate of firing remains at that level until the light is switched off, at which point it drops sharply and then returns to the background level, a mirror image of what happened when the light was switched on.

In connection with the main theme of this paper, a point of interest is that the positive spike when the light is switched on, and the negative spike when the light is switched off, have the effect of marking boundaries, first between dark and light, and later between light and dark. In effect, this is a form of run-length coding (Section 2.1.4). At the first boundary, the positive spike marks the fact of the light coming on. As long as the light stays on, there is no need for that information to be constantly repeated, so there is no need for the rate of firing to remain at a high level. Likewise, when the light is switched off, the negative spike marks the transition to darkness and, as before, there is no need for constant repetition of information about the new low level of illumination.[14]

Another point of interest is that this pattern of responding—adaptation to constant stimulation—can be explained via the action of inhibitory nerve fibres that bring the rate of firing back to the background rate when there is little or no variation in the sensory input [92].

Inhibitory mechanisms are widespread in the brain [84, p. 45] and it appears that, in general, their role is to reduce or eliminate redundancies in information ([109, Section 9]), in keeping with the main theme of this paper.

14 Other examples of adaptation

Adaptation is also evident at the level of conscious awareness. If, for example, a fan starts working nearby, we may notice the hum at first but then adapt to the sound and cease to be aware of it. But when the fan stops, we are likely to notice the new quietness at first but adapt again and stop noticing it.

Another example is the contrast between how we become aware if something or someone touches us but we are mostly unaware of how our clothes touch us in many places all day long. We are sensitive to something new and different and we are relatively insensitive to things that are repeated.

As with adaptation in the eye of *Limulus*, these other kinds of adaptation

[14] It is recognised that this kind of adaptation in eyes is a likely reason for small eye movements when we are looking at something, including sudden small shifts in position ('microsaccades'), drift in the direction of gaze, and tremor [57]. Without those movements, there would be an unvarying image on the retina so that, via adaptation, what we are looking at would soon disappear!

may be seen as examples of the run-length coding technique for compression of information.

15 Discovering the segmental structure of language

There is evidence that much of the segmental structure of language—words and phrases—may be discovered via ICMUP, as described in the following two subsections. To the extent that these mechanisms model aspects of HLPC, they provide evidence for ICHLPC.

With regard to Section 2.4, about the possible role of quantification in empirical evidence for ICHLPC, the MK10 Computer Model, designed for the discovery of segmental structure in language and outlined below, assigns a central role to the quantification of frequencies with which basic symbols such as letters, or sequences of symbols, occur in any given sample of language.

15.1 The word structure of natural language

As can be seen in Figure 13, people normally speak in 'ribbons' of sound, without gaps between words or other consistent markers of the boundaries between words. In the figure—the waveform for a recording of the spoken phrase "on our website"—it is not obvious where the word "on" ends and the word "our" begins, and likewise for the words "our" and "website". Just to confuse matters, there are three places within the word "website" that look as if they might be word boundaries.

Given that words are not clearly marked in the speech that young children hear, how do they get to know that language is composed of words? Learning to read could provide an answer but it appears that young children develop an understanding that language is composed of words well before the age when, normally, they are introduced to reading. Perhaps more to the point is that there are still, regrettably, many children throughout the world that are never introduced to reading but, in learning to talk and to understand speech, they inevitably develop a knowledge of the structure of language, including words.[15]

[15] It has been recognised for some time that skilled speakers of any language have an ability to create or recognise sentences that are grammatical but new to the world. Chomsky's well-known example of such a sentence is *Colorless green ideas sleep furiously.* [25, p. 15], which, when it was first published, was undoubtedly novel. This ability to create or recognise grammatical but novel sentences implies that knowledge of a language means knowledge of words as discrete entities that can form novel combinations.

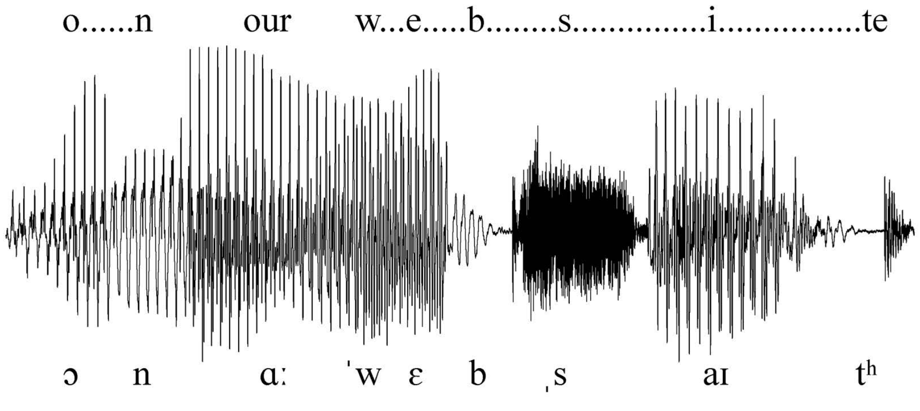


Figure 13: Waveform for the spoken phrase "On our website" with an alphabetic transcription above the waveform and a phonetic transcription below it. With thanks to Sidney Wood of SWPhonetics (swphonetics.com) for the figure and for permission to reproduce it.

In keeping with the main theme of this paper, ICMUP provides an answer [99, 100, 103] which works largely via ICMUP and can reveal much of the word structure in an English-language text from which all spaces and punctuation has been removed [107, Section 5.2]. It is true that there are added complications with speech but it seems likely that similar principles apply.

This discovery of word structure by the MK10 program, illustrated in Figure 14, is achieved without the aid of any kind of externally-supplied dictionary or other information about the structure of English. The program builds its own dictionary via 'unsupervised' learning using only the unsegmented sample of English with which it is supplied. It learns without the assistance of any kind of 'teacher', or data that is marked as 'wrong', or the grading of samples from simple to complex (*cf.* [36]).

Statistical tests show that the correspondence between the computer-assigned word structure and the original (human) division into words is significantly better than chance.

Two aspects of the MK10 model strengthen its position as a model of what children do in learning the segmental structure of language [100]: the growth in the lengths of words learned by the program corresponds quite well with the same measure for children; and the pattern of changing numbers of new words that are learned by the program at different stages corresponds quite well with the equivalent pattern for children.

Discovering the word structure of language via ICMUP is another example

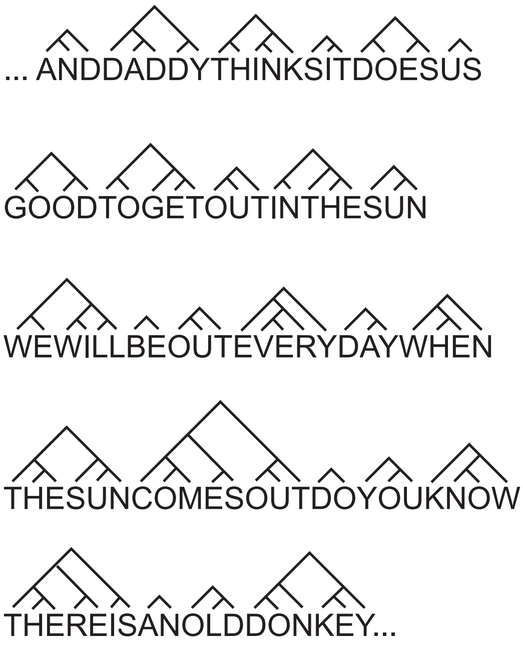


Figure 14: Part of a parsing created by the MK10 Computer Model [100] from a 10,000 letter sample of English (book 8A of the Ladybird Reading Series) with all spaces and punctuation removed. The program derived this parsing from the sample alone, without any prior dictionary or other knowledge of the structure of English. Reproduced from Figure 7.3 in [103], with permission.

of the DONSVIC principle, mentioned in Section 12—because words are the kinds of 'natural' structure which are the subject of the DONSVIC principle, and because ICMUP provides a key to how they may be discovered.

15.2 The phrase structure of natural language

In addition to its achievements in learning the word structure of natural language, the MK10 Computer Model, featured in Section 15.1, does quite a good job at discovering the phrase structure of unsegmented text in which each word has been replaced by a symbol representing the grammatical class of the word [101, 103]. An example is shown in Figure 15. As before, the program works without any prior knowledge of the structure of English and, apart from the initial assignment of word classes, it works in unsupervised mode without the assistance of any kind of 'teacher', or anything equivalent. As before, statistical tests show that the correspondence between computer-assigned and human-assigned structures is statistically significant.[16]

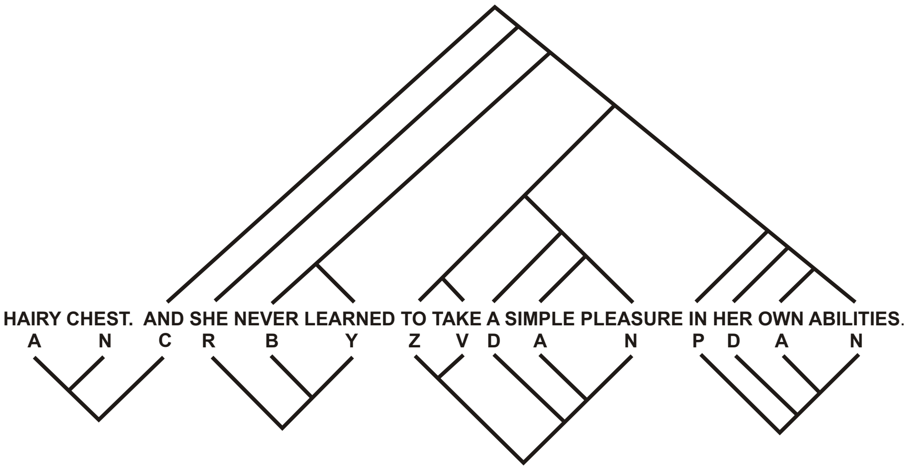

HAIRY CHEST. AND SHE NEVER LEARNED TO TAKE A SIMPLE PLEASURE IN HER OWN ABILITIES.
A N C R B Y Z V D A N P D A N

Figure 15: One sentence from a 7600 word sample from the book *Jerusalem the Golden* (by Margaret Drabble) showing (above the text) a surface structure analysis, and (below the text) the parsing developed by the MK10 Computer Model at a late stage of processing [101]. This figure is reproduced by kind permission of Kingston Press Services Ltd.

Since ICMUP is central in the workings of the MK10 Computer Model, this result suggests that ICMUP may have a role to play, not merely in dis-

[16] Thanks to Dr. Isabel Forbes, a person qualified in theoretical linguistics, for the assignment of grammatical class symbols to words in the given text, and for phrase-structure analyses of the text.

covering the phrase structure of language, but more generally in discovering the grammatical structure of language.

16 Grammatical inference

Regarding the last point from the previous section, it seems likely that learning the grammar of a language may also be understood in terms of ICMUP. Evidence in support of that expectation comes from research with two programs designed for grammatical inference:

- *The SNPR Computer Model.* The SNPR Computer Model, which was developed from the MK10 Computer Model, can discover plausible grammars from samples of English-like artificial languages [102, 103]. This includes the discovery of segmental structures, classes of structure, and abstract patterns. ICMUP is central in how the program works.

- *The SP Computer Model.* The SP Computer Model, one of the main products of the SP programme of research, achieves results at a similar level to that of SNPR. As before, ICMUP is central in how the program works. With the solution of some residual problems, outlined in [107, Section 3.3], there seems to be a real possibility that the SP System will be able to discover plausible grammars from samples of natural language. Also, it is anticipated that, with further development, the program may be applied to the learning of non-syntactic 'semantic' knowledge, and the learning of grammars in which syntax and semantics are integrated.

What was the point of developing the SP Computer Model when it does no better at grammatical inference than the SNPR Computer Model? The reason is that the SNPR Computer Model, which was designed for the discovery of syntactic structures and worked mainly via the building of hierarchical structures, was not compatible with the new and much more ambitious goal of the SP programme of research: to simplify and integrate observations and concepts across artificial intelligence, mainstream computing, mathematics, and HLPC. What was needed was a new organising principle that would accommodate hierarchical structures and several other kinds of structure as well.

It turns out that the SP-multiple-alignment concept is much more versatile than the hierarchical organising principle in the SNPR program, providing for several aspects of intelligence and the representation and processing of a variety of knowledge structures of which hierarchical structures is only one (Section 2.2.5). It appears that the SP System provides a much firmer foundation for the development of human-level intelligence than the SNPR Computer Model or indeed deep learning models, as discussed in [110, Section V].

With regard to Section 2.4 about the possible role of quantification in empirical evidence for ICHLPC, the SNPR Computer Model and the SP Computer Model, like the MK10 Computer Model (Section 15), both have a central role for quantification of the frequencies with which basic symbols such as letters, or contiguous or broken patterns of symbols, occur in any given sample of data.

17 Generalisation, the correction of wrong generalisations, and 'dirty data'

Issues relating to generalisation in learning are best described with reference to the Venn diagram shown in Figure 16. That figure relates to the unsupervised learning of a natural language but it appears that generalisation issues in other areas of learning are much the same.

The evidence to be described derives largely from the SNPR Computer Model and the SP Computer Model. Since both models are founded on ICMUP, evidence that they have human-like capabilities with generalisation and related phenomena may be seen as evidence in support of ICHLPC.

In the figure, the smallest envelope shows the finite but large sample of 'utterances' from which a young child learns his or her native language[17] (which we shall call **L**)—where an 'utterance' is a speech sound of any kind, and the speakers from which a young child learns are adults or older children. The middle-sized envelope shows the (infinite) set of utterances in **L**, and the largest envelope shows the (infinite) set of all possible utterances, including those that are in **L** and those which are not. 'Dirty data' are the many 'ungrammatical' utterances that children normally hear—outside the envelope for **L** but inside the envelope representing the utterances from which a young child learns.

The child generalises 'correctly' when he or she infers **L**, and only **L**, from the finite sample he or she has heard, including dirty data. Anything that spills over into the outer envelope, like "mouses" as the plural of "mouse" or "buyed" as the past tense of "buy", is an over-generalisation, while failure to learn the whole of **L** represents under-generalisation.

[17] To keep things simple in this discussion we shall assume that each child learns only one first language, although many children learn two or more first languages.

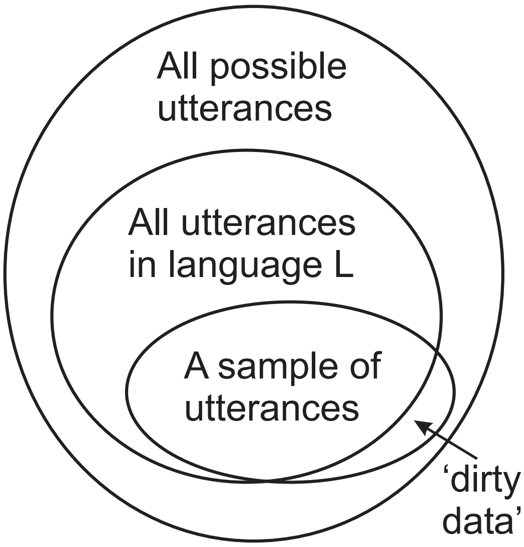


Figure 16: Categories of utterances involved in the learning of a first language, **L**. In ascending order size, they are: the finite sample of utterances from which a child learns; the (infinite) set of utterances in **L**; and the (infinite) set of all possible utterances. Adapted from Figure 7.1 in [103], with permission.

In connection with the foregoing summary of concepts relating to generalisation, there are three main problems:

- *Generalisation without over-generalisation.* How can we generalise our knowledge without over-generalisation, and this in the face of evidence that children can learn their first language or languages without the correction of errors by parents or teachers or anything equivalent?[18]

- *Generalisation without under-generalisation.* How can we generalise our knowledge without under-generalisation? As before, there is evidence that learning of a language can be achieved without explicit teaching.

- *Dirty data.* How can we learn correct knowledge despite errors in the

[18] Evidence comes chiefly from children who learned language without the possibility that anyone might correct their errors. Christy Brown was a cerebral-palsied child who not only lacked any ability to speak but whose bodily handicap was so severe that for much of his childhood he was unable to demonstrate that he had normal comprehension of speech and non-verbal forms of communication [17]. Hence, his learning of language must have been achieved without the possibility that anyone might correct errors in his spoken language.

examples we hear. Again, it appears that this can be done without correction of errors.

These things are discussed quite fully in [105, Section 9.5.3] and [107, Section 5.3]. There is also relevant discussion in [110, Section V-H and XI-C].

In brief, IC provides an answer to all three problems like this: for a given body of raw data, **I**, compress it thoroughly via unsupervised learning; the resulting compressed version of **I** may be split into two parts, a *grammar* and an *encoding* of **I** in terms of the grammar; normally, the grammar generalises correctly without over- or under-generalisation, and errors in **I** are weeded out; the encoding may be discarded.

This scheme is admirably simple, but, so far, the evidence in support of it is only informal, derived largely from informal experiments with English-like artificial languages with the SNPR Computer Model of language learning ([102], [103]) and the SP Computer Model [105, Section 9.5.3].

The weeding out of errors via this scheme may seem puzzling, but errors, by their nature, are rare. The grammar retains the repeating parts of **I** (which are relatively common), while the encoding contains the non-repeating parts including most of the errors. 'Errors' which are not rare acquire the status of 'dialect' and cease to be regarded as errors.

A problem with research in this area is that the identification of any over- or under-generalisations produced by the above scheme or any other model depends largely on human intuitions. But this is not so very different from the long-established practice in research on linguistics of using human judgements of grammaticality to establish what any given person knows about a particular language.

The problem of generalising our learning without over- or under-generalisation applies to the learning of a natural language and also to the learning of such things as visual images. It appears that the solution outlined here has distinct advantages compared with, for example, what appear to be largely *ad hoc* solutions that have been proposed for deep learning in artificial neural networks [110, Section V-H].

As noted above, evidence for human-like generalisation with the SNPR and SP computer models, without either over- or under-generalisation, may be seen as evidence in support of ICMUP as a unifying principle in HLPC.

18 Perceptual constancies

It has long been recognised that our perceptions are governed by *constancies*:

- *Size constancy.* To a large extent, we judge the size of an object to be constant despite wide variations in the size of its image on the retina [31, pp. 40-41].
- *Lightness constancy.* We judge the lightness of an object to be constant despite wide variations in the intensity of its illumination [31, p. 376].
- *Colour constancy.* We judge the colour of an object to be constant despite wide variations in the colour of its illumination [31, p. 402].

These kinds of constancy, and others such as shape constancy and location constancy, may each be seen as a means of encoding information economically: it is simpler to remember that a particular person is "about my height" than many different judgements of size, depending on how far away that person is. In a similar way, it is simpler to remember that a particular object is "black" or "red" than all the complexity of how its lightness or its colour changes in different lighting conditions.

By filtering out variations due to viewing distance or the intensity or colour of incident light, we can facilitate ICMUP and thus, for example, in watching a football match, simplify the process of establishing that there is (normally) just one ball on the pitch and not many different balls depending on viewing distances, whether the ball is in a bright or shaded part of the pitch, and so on.

19 Kinds of redundancy that people find difficult or impossible to detect

Although the matching and unification of patterns is often effective in the detection and reduction of redundancy in information, there are kinds of redundancy that are not easily revealed via ICMUP. It seems that those kinds of redundancy are also ones that people find difficult or impossible to detect. A well-known example is the decimal representation of π, which appears to most people to be entirely random, but which can be created by a simple program so that, in terms of Algorithmic Information Theory, it contains much redundancy.

At first sight, this observation seems to contradict the main thesis of this paper, that much of HLPC may be may be understood as IC. But there is nothing in the ICHLPC thesis to say that people can or should be able to detect all kinds of redundancy via ICMUP. And the apparent randomness of the decimal representation of π suggests that any natural or artificial system that works via ICMUP would fail to detect the redundancy in data of that kind.

In short, what appears at first sight to be evidence against ICHLPC, turns out to be evidence in support of that thesis: the failure of most people to detect the redundancy in the decimal representation of π may be explained via the ICHLPC thesis, together with the apparent weakness of ICMUP in discovering and reducing that kind of redundancy.

20 Mathematics

A discussion of mathematics may seem out of place in a paper about ICHLPC but mathematics is relevant because it has been developed over many years as an aid to human thinking. For that reason, in the spirit of George Boole's *An investigation of the laws of thought* [15],[19] a consideration of the organisation and workings of mathematics is relevant to ICHLPC.

In [113] it is argued that much of mathematics, perhaps all of it, may be seen as a set of techniques for the compression of information via the matching and unification of patterns, and their application. In case this seems implausible:

- *An equation as a compressed representation of data.* An equation like Albert Einstein's $E = mc^2$ may be seen as a very compressed representation of what may be a very large set of data points relating energy (E) and mass (m), with the speed of light (c) as a constant. Similar things may be said about such well-known equations as $s = (gt^2)/2$ (derived from Newton's second law of motion), $a^2 + b^2 = c^2$ (Pythagoras's equation), $PV = k$ (Boyle's law), and $F = q(E + v \times B)$ (the charged-particle equation).

- *Variants of ICMUP may be seen at work in mathematical notations.* The second, third, and fourth of the variants of ICMUP outlined in Section 2.1 may be seen at work in mathematical notations. For example: multiplication as repeated addition may be seen as an example of run-length coding.

Owing to the close connections between logic and mathematics, and between computing and mathematics, it seems likely that similar principles apply in logic and in computing [113, Section 4].

[19] And perhaps also in the spirit of William Thomson's *Outline of the Laws of Thought* [88], although his orientation is more towards concepts in logic than mathematics.

Although in this research it has seemed necessary to avoid too much dependence on mathematics (for reasons outlined in Section 2.3), there is now the interesting possibility that the scope of mathematics may be greatly extended by incorporating within it such concepts as SP-multiple-alignment and other elements of the SP Theory [113, Section 7].

21 Evidence for ICHLPC via the SP System

Another strand of empirical evidence for ICHLPC is via the SP System and the central role within it of SP-multiple-alignment (Section 2.2.2), a variant of ICMUP which, as described in Section 2.1.7, encompasses the six others described in Section 2.1.

The evidence for ICHLPC via the SP System derives largely from the strengths of the SP System in modelling several aspects of HLPC, summarised in Section 2.2.5, and described in progressively more detail in [112, Sections 4, 5, and 6], in [107], and in [105].

22 Some apparent contradictions and how they may be resolved

The idea that IC is fundamental in HLPC, and also in the SP Theory as a theory of HLPC, seems to be contradicted by:

- The ways in which people may create redundant copies of information as well as how they may compress information;

- The fact that redundancy in information is often useful in detecting and correcting errors, and in the storage and processing of information;

- A less direct challenge to ICHLPC, and the SP Theory as a theory of HLPC, is persuasive evidence, described by Gary Marcus [54], that in many respects, the human mind is a kluge, meaning “a clumsy or inelegant—yet surprisingly effective—solution to a problem” (p 2).

These apparent contradictions and how they may be resolved are discussed in Appendix C.

23 Conclusion

This paper presents evidence for the idea that much of human learning, perception, and cognition (HLPC), may be understood as IC, often via the matching and unification of patterns.

The paper is part of a programme of research developing the *SP Theory of Intelligence* and its realisation in the *SP Computer Model*—a theory which aims to simplify and integrate observations and concepts across artificial intelligence, mainstream computing, mathematics, and HLPC.

Since IC is central in the SP Theory, evidence for IC in HLPC, presented in this paper in Sections 4 to 20 inclusive (but excluding Section 21), strengthens empirical support for the SP Theory, viewed as a theory of HLPC.

More direct empirical evidence for the SP Theory as a theory of HLPC—summarised in Section 2.2.5—provides evidence for the IC in HLPC thesis which is additional to that in Sections 4 to 20 inclusive.

Four possible objections to the IC in HLPC thesis, and the SP Theory, are described in Appendix C, with answers to those objections.

The ideas developed in this research may be seen to be part of a 'Big Picture' of the importance of IC in at least six areas, outlined in Section 2.6.

Acknowledgements

I am grateful to anonymous reviewers for helpful, constructive comments on earlier versions of this paper. Thanks also to Dr. Isabel Forbes for the assignment of grammatical class symbols to words in a given text, and for phrase-structure analyses, as described in Section 15.2.

Appendices

A Mathematics associated with ICMUP, and mathematics incorporated in the SP System

As mentioned in Section 2.3, this appendix details some mathematics associated with ICMUP, and some of the mathematics incorporated in the SP System.

A.1 Searching for repeating patterns

At first sight, the process of searching for repeating patterns (Sections 2.1.1 and 2.2.2) is simply a matter of comparing one pattern with another to see whether they match each other or not. But there are, typically, many alternative ways in which patterns within a given body of information, **I**, may be compared—and some are better than others. We are interested in finding those matches between patterns that, via unification, yield most compression—and a little reflection shows that this is not a trivial problem [105, Section 2.2.8.4].

Maximising the amount of redundancy found means maximising R where:

$$R = \sum_{i=1}^{i=n} (f_i - 1) \cdot s_i, \tag{1}$$

f_i is the frequency of the ith member of a set of n patterns and s is its size in bits. Patterns that are both big and frequent are best. This equation applies irrespective of whether the patterns are coherent substrings or patterns that are discontinuous within **I**.

Maximising R means searching the space of possible unifications for the set of big, frequent patterns that gives the best value. For a sequence containing N symbols, the number of possible subsequences (including single symbols and all composite patterns, both coherent and fragmented) is:

$$P = 2^N - 1. \tag{2}$$

The number of possible comparisons is the number of possible pairings of subsequences which is:

$$C = P(P - 1)/2. \tag{3}$$

For all except the very smallest values of N, the value of P is very large and the corresponding value of C is huge. In short, the abstract space of possible comparisons between patterns and thus the space of possible unifications is, in the great majority of cases, astronomically large.

Since the space is normally so large, it is not feasible to search it exhaustively. For that reason, we cannot normally guarantee to find the theoretically ideal answer, and normally we cannot know whether or not we have found the theoretically ideal answer.

In general, we need to use heuristic methods in searching—conducting the search in stages and discarding all but the best results at the end of each stage—and we must be content with answers that are “reasonably good”.

A.2 Information, compression of information, inductive inference and probabilities

Solomonoff [82] seems to have been one of the first people to recognise the close connection that exists between IC and *inductive inference* (Section 2.5): predicting the future from the past, and calculating probabilities for such inferences. The connection between them—which may at first sight seem obscure—lies in the redundancy-as-repetition-of-patterns view of redundancy and how that relates to IC (Section 2.1, [105, Section 2.2.11]):

- Patterns that repeat within **I** represent redundancy in **I**, and IC can be achieved by reducing multiple instances of any pattern to one.
- When we make inductive predictions about the future, we do so on the basis of repeating patterns. For example, the repeating pattern 'Spring, Summer, Autumn, Winter' enables us to predict that, if it is Spring time now, Summer will follow.

Thus IC and inductive inference are closely related to concepts of frequency and probability. Here are some of the ways in which these concepts are related:

- Probability has a key role in Shannon's concept of information. In that perspective, the average quantity of information conveyed by one symbol in a sequence is:

  $$H = -\sum_{i=1}^{i=n} p_i \log p_i, \qquad (4)$$

  where p_i is the probability of the ith type in the alphabet of n available alphabetic symbol types. If the base for the logarithm is 2, then the information is measured in 'bits'.
- Measures of frequency or probability are central in techniques for economical coding such as the Huffman method [26, Section 5.6] or the Shannon-Fano-Elias method [26, Section 5.9].
- In the redundancy-as-repetition-of-patterns view of redundancy and IC, the frequencies of occurrence of patterns in **I** is a main factor (with the sizes of patterns) that determines how much compression can be achieved.

- Given a body of (binary) data that has been 'fully' compressed (so that it may be regarded as random or nearly so), its absolute probability may be calculated as $p_{ABS} = 2^{-L}$, where L is the length (in bits) of the compressed data.

Probability and IC may be regarded as two sides of the same coin. That said, they provide different perspectives on a range of problems. In this research, the IC perspective—with redundancy-as-repetition-of-patterns—seems to be more fruitful than viewing the same problems through the lens of probability. In the first case, one can see relatively clearly how compression may be achieved by the primitive operation of unifying patterns whereas these ideas are obscured when the focus is on probabilities.

A.3 Random-dot stereograms

A particularly clear example of the kind of search described in Appendix A.1 is what the brain has to do to enable one to see the figure in the kinds of random-dot stereogram described in Section 11.

In this case, assuming the left image has the same number of pixels as the right image, the size of the search space is:

$$S = P^2/2 \tag{5}$$

where P is the number of possible patterns in each image, calculated in the same way as was described in Appendix A.1. The fact that the images are two dimensional needs no special provision because the original equations cover all combinations of atomic symbols.

For any stereogram with a realistic number of pixels, this space is very large indeed. Even with the very large processing power represented by the 10^{11} neurons in the brain, it is inconceivable that this space can be searched in a few seconds and to such good effect without the use of heuristic methods.

David Marr [55, Chapter 3] describes two algorithms that solve this problem. In line with what has just been said, both algorithms rely on constraints on the search space and both may be seen as incremental search guided by redundancy-related metrics.

A.4 Coding and the evaluation of SP-multiple-alignments in terms of IC

Given an SP-multiple-alignment like one of the two shown in Figure 4 (Section 2.2.2), one can derive a *code SP-pattern* from the SP-multiple-alignment in the following way:

1. Scan the SP-multiple-alignment from left to right looking for columns that contain an SP-symbol by itself, not aligned with any other symbol.
2. Copy these SP-symbols into a code pattern in the same order that they appear in the SP-multiple-alignment.

The code SP-pattern derived in this way from the SP-multiple-alignment shown in Figure 4 is '`S 0 2 4 3 7 6 1 #S`'. This is, in effect, a compressed representation of those symbols in the New pattern that form hits with Old symbols in the SP-multiple-alignment.

Given a code SP-pattern derived in this way, we may calculate a 'compression difference' as:

$$CD = B_N - B_E \tag{6}$$

or a 'compression ratio' as:

$$CR = B_N / B_E, \tag{7}$$

where B_N is the total number of bits in those symbols in the New pattern that form hits with Old symbols in the SP-multiple-alignment, and B_E is the total number of bits in the code SP-pattern (the 'encoding') that has been derived from the SP-multiple-alignment as described above.

In each of these equations, B_N is calculated as:

$$B_N = \sum_{i=1}^{h} C_i, \tag{8}$$

where C_i is the size of the code for ith symbol in a sequence, $H_1 ... H_h$, comprising those symbols within the New pattern that form hits with Old symbols within the SP-multiple-alignment (Appendix A.5).

B_E is calculated as:

$$B_E = \sum_{i=1}^{s} C_i, \tag{9}$$

where C_i is the size of the code for ith symbol in the sequence of s symbols in the code pattern derived from the SP-multiple-alignment (Appendix A.5).

A.5 Encoding individual symbols

The simplest way to encode individual symbols in the New pattern and the set of Old patterns in an SP-multiple-alignment is with a 'block' code using a fixed number of bits for each symbol. But the SP Computer Model uses variable-length codes for symbols, assigned in accordance with the Shannon-Fano-Elias coding scheme [26, Section 5.9] so that the shortest codes represent the most frequent alphabetic symbol types and *vice versa*. Although this scheme is slightly less efficient than the well-known Huffman scheme, it has been adopted because it avoids some anomalous results that can arise with the Huffman scheme.

For the Shannon-Fano-Elias calculation, the frequency of each alphabetic symbol type (f_{st}) is calculated as:

$$f_{st} = \sum_{i=1}^{P} (f_i \times o_i) \tag{10}$$

where f_i is the (notional) frequency of the ith pattern in the collection of Old SP-patterns (the *grammar*) used in the creation of the given SP-multiple-alignment, o_i is the number of occurrences of the given symbol in the ith SP-pattern in the grammar and P is the number of SP-patterns in the grammar.

A.6 Calculation of probabilities associated with any given SP-multiple-alignment

As may be seen in [105, Chapter 7], the formation of SP-multiple-alignments in the SP framework supports a variety of kinds of probabilistic reasoning. The core idea is that any Old symbol in a SP-multiple-alignment that is *not* aligned with a New symbol represents an inference that may be drawn from the SP-multiple-alignment. This section describes how absolute and relative probabilities for such inferences may be calculated.

A.6.1 Absolute probabilities

Any sequence of L symbols, drawn from an alphabet of $|A|$ alphabetic types, represents one point in a set of N points where N is calculated as:

$$N = |A|^L. \tag{11}$$

If we assume that the sequence is random or nearly so, which means that the N points are equi-probable or nearly so, the probability of any one point (which represents a sequence of length L) is close to:

$$p_{ABS} = |A|^{-L}. \tag{12}$$

In the SP Computer Model, the value of $|A|$ is 2.

This equation may be used to calculate the absolute probability of the code, C, derived from the SP-multiple-alignment as described in Appendix

A.4. p_{ABS} may also be regarded as the absolute probability of any inferences that may be drawn from the SP-multiple-alignment as described in [105, Section 7.2.2].

A.6.2 Relative probabilities

The absolute probabilities of SP-multiple-alignments, calculated as described in the last subsection, are normally very small and not very interesting in themselves. From the standpoint of practical applications, we are normally interested in the *relative* values of probabilities, not their *absolute* values.

The procedure for calculating relative values for probabilities (p_{REL}) is as follows:

1. For the SP-multiple-alignment which has the highest CD (which we shall call the *reference SP-multiple-alignment*), identify the symbols from New which are encoded by the SP-multiple-alignment. We will call these symbols the *reference set of symbols in New.*

2. Compile a *reference set of SP-multiple-alignments* which includes *the SP-multiple-alignment with the highest CD and all other SP-multiple-alignments (if any) which encode exactly the reference set of symbols from New, neither more nor less.*

3. The SP-multiple-alignments in the reference set are examined to find and remove any rows which are redundant in the sense that all the symbols appearing in a given row also appear in another row in the same order.[20] Any SP-multiple-alignment which, after editing, matches another SP-multiple-alignment in the set is removed from the set.

4. Calculate the sum of the values for p_{ABS} in the reference set of SP-multiple-alignments:

$$p_{A\_SUM} = \sum_{i=1}^{i=R} p_{ABS_i} \tag{13}$$

   where R is the size of the reference set of SP-multiple-alignments and p_{ABS_i} is the value of p_{ABS} for the ith SP-multiple-alignment in the reference set.

[20] If Old is well compressed, this kind of redundancy amongst the rows of a SP-multiple-alignment should not appear very often.

5. For each SP-multiple-alignment in the reference set, calculate its relative probability as:

$$p_{REL_i} = p_{ABS_i} / p_{A\_SUM}. \tag{14}$$

The values of p_{REL} calculated as just described seem to provide an effective means of comparing the SP-multiple-alignments in the reference set. Normally, this will be those SP-multiple-alignments which encode the same set of symbols from New as the SP-multiple-alignment which has the best overall CD.

A.7 Sifting and sorting of SP-patterns in unsupervised learning in the SP System

In the process of unsupervised learning in the SP System (Section 2.2.3, [105, Chapter 9]), which starts with a set of New SP-patterns, there is a process of sifting and sorting Old SP-patterns that are created by the SP System to develop one or more alternative collections of Old SP-patterns (*grammars*), each one of which scores well in terms of its capacity for the economical encoding of the given set of New SP-patterns.

When all the New SP-patterns have been processed in this way, there is a set A of full SP-multiple-alignments, divided into $b_1 ... b_m$ disjoint subsets, one for each SP-pattern from the given set of New SP-patterns. From these SP-multiple-alignments, the program computes the frequency of occurrence of each of the $p_1 ... p_n$ Old SP-patterns as:

$$f_i = \sum_{j=1}^{j=m} max(p_i, b_j) \tag{15}$$

where $max(p_i, b_j)$ is the maximum number of times that p_i appears in any *one* of the SP-multiple-alignment in the subset b_j.

The program also compiles an alphabet of the alphabetic symbol types, $s_1 ... s_r$, in the Old SP-patterns and, following the principles just described, computes the frequency of occurrence of each alphabetic symbol type as:

$$F_i = \sum_{j=1}^{j=m} max(s_i, b_j) \tag{16}$$

where $max(s_i, b_j)$ is the maximum number of times that s_i appears in any *one* SP-multiple-alignment in subset b_j. From these values, the encoding cost of each alphabetic symbol type is computed using the Shannon-Fano-Elias method as before [26, Section 5.9].

In the process of building alternative grammars, the tree of such alternatives is pruned periodically to keep it within reasonable bounds. Values for G, E and $(G + E)$ (which we will refer to as T) are calculated for each grammar and, at each stage, grammars with high values for T are eliminated.

For a given grammar comprising SP-patterns $p_1 ... p_g$, the value of G is calculated as:

$$G = \sum_{i=1}^{i=g} (\sum_{j=1}^{j=L_i} s_j) \tag{17}$$

where L_i is the number of symbols in the ith SP-pattern and s_j is the encoding cost of the jth SP-symbol in that SP-pattern.

Given that each grammar is derived from a set $a_1 ... a_n$ of SP-multiple-alignments (one SP-multiple-alignment for each pattern from New), the value of E for the grammar is calculated as:

$$E = \sum_{i=1}^{i=n} e_i \tag{18}$$

where e_i is the size, in bits, of the code SP-pattern derived from the ith SP-multiple-alignment.

A.8 Finding good matches between two sequences of symbols

At the heart of the SP Computer Model is a process for finding good matches between two sequences of symbols, mentioned in Section 2.2.2 and described quite fully in [105, Appendix A]. What has been developed is a version of dynamic programming with the advantage that it can find two or more good matches between sequences, not just one good match.

The search process uses a measure of probability, p_n, as its metric. This metric provides a means of guiding the search which is effective in practice and appears to have a sound theoretical basis. To define p_n and to justify it theoretically, it is necessary first to define the terms and variables on which it is based:

- A sequence of matches between two sequences, sequence1 and sequence2, is called a 'hit sequence'.
- For each hit sequence $h_1 ... h_n$, there is a corresponding series of *gaps*, $g_1 ... g_n$. For any one hit, the corresponding gap is $g = g_q + g_d$, where g_q is the number of unmatched characters in the query between the

query character for the given hit in the series and the query character for the immediately preceding hit; and g_d is the equivalent gap in the database, g_1 is taken to be 0.

- A is the size of the *alphabet* of symbol types used in sequence1 and sequence2.
- p_1 is the probability of a match between any one symbol in sequence1 and any one symbol in sequence2 on the null hypothesis that all hits are equally probable at all locations. Its value is calculated as: $p_1 = 1/A$.

Using these definitions, the probability of any hit sequence of length n is calculated as:

$$p_n = \prod_{i=1}^{i=n} (1 - (1 - p_1)^{g_i+1}), \quad g_1 = 0$$

.

With this equation, is relatively easy to calculate the probability of the hit sequence up to and including any hit by using the stored value of the hit sequence up to and including the immediately preceding hit.

B Barlow's change of view about the significance of IC in mammalian learning, perception, and cognition, with comments

As noted in Section 3.1.1, Horace Barlow [11, p. 242] argued that "... the [compression] idea was right in drawing attention to the importance of redundancy in sensory messages ... but it was wrong in emphasizing the main technical use for redundancy, which is compressive coding." His main arguments follow, with my comments after each one, flagged with 'JGW'.

B.1 "Redundancy is not something useless that can be stripped off and ignored"

> "It is important to realize that redundancy is not something useless that can be stripped off and ignored. An animal must identify what is redundant in its sensory messages, for this can tell it about structure and statistical regularity in its environment that are important for its survival." [11, p. 243], and "It is ... knowledge and recognition of ... redundancy, not its reduction, that matters." [11, p. 244].

JGW: Barlow is right to say that knowledge of and recognition of redundancy is important "for this can tell [an animal] about structure and statistical regularity in its environment that are important for its survival.". In keeping with that remark, knowledge of the frequency of occurrence of any pattern may serve in the calculation of absolute and relative probabilities ([105, Section 3.7], [107, Section 4.4]) and it can be the key to the correction of errors, as Barlow mentions in the quote from him in the heading of Appendix B.2.

But in the SP System, redundancy is not treated as "something useless that can be stripped off and ignored". Patterns that repeat are reduced to a single instance and the frequency of occurrence of that single instance is recorded. The existence of single instances like that, each with a record of its frequency of occurrence, is very important, both in the way that the SP System builds its model of the world, and also in the way that it makes inferences and calculates probabilities of those inferences.

As noted in Section 10, if we did not compress sensory information, "our brains would quickly become cluttered with millions of copies of things that we see around us—people, furniture, cups, trees, and so on—and likewise for sounds and other sensory inputs." And as noted in Section 3.1.1, Barlow himself has pointed out that the mismatch between the relatively large amounts of information falling on the retina and the relatively small transmission capacity of the optic nerve suggests that sensory information is likely to be compressed [8, p. 548]. And he has also pointed out that in animals like cats, monkeys, and humans, "one obvious type of redundancy in the messages reaching the brain is the very nearly exact reduplication of one eye's message by the other eye" [9, p. 213], and because we normally see one view, not two, the duplication implies that the two views are merged and thus compressed. In general, the evidence presented in Sections 4 to 21 points strongly to IC as a prominent feature of HLPC.

B.2 "Redundancy is mainly useful for error avoidance and correction"

JGW: The heading above, from [11, p. 244], implies that compression of information via the reduction of redundancy is relatively unimportant, in keeping with the quotes from Barlow in the previous subsection.

Redundancy can certainly be useful in the avoidance of or correction of errors (Appendix C.2). But experience in the development and application of the SP Computer Model has shown that compression of information via the reduction of redundancy is also needed for such tasks as the parsing of natural language, pattern recognition, and grammatical inference. And compression of information may on occasion be intimately related to the correction of errors of omission, commission, and substitution, as described in Appendix C.2 and illustrated in Figure 20 (see also [107, Section 4.2.2] and [105, Section 6.2]).

B.3 "There are very many more neurons at higher levels in the brain" and "compressed, non-redundant, representation would not be at all suitable for the kinds of task that brains have to perform"

Following the remark that "This is the point on which my own opinion has changed most, partly in response to criticism, partly in response to new facts that have emerged." [11, p. 244], Barlow writes:

> "Originally both Attneave and I strongly emphasized the economy that could be achieved by recoding sensory messages to take advantage of their redundancy, but two points have become clear since those early days. First, anatomical evidence shows that there are very many more neurons at higher levels in the brain, suggesting that redundancy does not decrease, but actually increases. Second, the obvious forms of compressed, non-redundant, representation would not be at all suitable for the kinds of task that brains have to perform with the information represented; ..." [11, pp. 244–245].

and

> "I think one has to recognize that the information capacity of the higher representations is likely to be greater than that of the representation in the retina or optic nerve. If this is so, redundancy must increase, not decrease, because information cannot be created." [11, p. 245].

JGW: There seem to be two problems here:

- The likelihood that there are "very many more neurons at higher levels in the brain [than at the sensory levels]" and that "the information capacity of the higher representations is likely to be greater than that of the representation in the retina or optic nerve" need not invalidate ICHLPC. It seems likely that many of the neurons at higher levels are concerned with the storage of one's accumulated knowledge over the period from one's birth to one's current age ([105, Chapter 11], [109, Section 4]). By contrast, neurons at the sensory level would be concerned only with the processing of sensory information at any one time.

Although knowledge in one's long-term memory stores is likely to be highly compressed and only a partial record of one's experiences, it is likely, for most of one's life except early childhood, to be very much larger than the sensory information one is processing at any one time. Hence, it should be no surprise to find many more neurons at higher levels than at the sensory level.

- For reasons given in Appendix B.4, next, there are reasons for doubting the proposition that "the obvious forms of compressed, non-redundant, representation would not be at all suitable for the kinds of task that brains have to perform with the information represented."

B.4 "Compressed representations are unsuitable for the brain"

Under the heading above, Barlow writes:

> "The typical result of a redundancy-reducing code would be to produce a distributed representation of the sensory input with a high activity ratio, in which many neurons are active simultaneously, and with high and nearly equal frequencies. It can be shown that, for one of the operations that is most essential in order to perform brain-like tasks, such high activity-ratio distributed representations are not only inconvenient, but also grossly inefficient from a statistical viewpoint ..." [11, p. 245].

JGW: With regard to these points:

- It is not clear why Barlow should assume that a redundancy-reducing code would, typically, produce a distributed representation, or that compressed representations are unsuitable for the brain. The SP System is dedicated to the creation of non-distributed compressed representations which work very well in several aspects of intelligence as outlined in Section 2.2.5 with pointers to where fuller information may be found. And in [109] it is argued that, in SP-Neural, such representations can be mapped on to plausible structures of neurons and their inter-connections that are quite similar to Donald Hebb's [37] concept of a 'cell assembly'.

- With regard to efficiency:
  - It is true that deep learning in artificial neural networks [76], with their distributed representations, are often hungry for computing resources, with the implication that they are inefficient. But otherwise they are quite successful with certain kinds of task, and there appears to be scope for increasing their efficiencies [24].
  - The SP System demonstrates that the compressed localist representations in the system are efficient and effective in a variety of kinds of task, as outlined in Section 2.2.5 with pointers to where fuller information may be found.

C Some apparent contradictions of ICHLPC and the SP Theory, and how they may be resolved

The apparent contradictions of ICHLPC, and the SP Theory as a theory of HLPC that were mentioned in Section 22, are discussed in the following three subsections, with suggested answers to those apparent contradictions.

C.1 Redundancy may be created by human brains, and via mathematics and computing

Any person may create redundancy by simply repeating any action, including any portion of speech or writing. Although this seems to contradict the ICHLPC thesis, the contradiction may be resolved as described in the following subsections.

C.1.1 Creating redundancy via IC

With a computer, it is very easy to create information containing large amounts of redundancy and to do it by a process which may itself be seen to entail the compression of information.

We can, for example, make a 'call' to the function defined in Figure 17, using the pattern '`oranges_and_lemons(100)`'. The effect of that call is to print out a highly redundant sequence containing 100 copies of the expression "Oranges and lemons, Say the bells of St. Clement's; ".

```
void oranges_and_lemons(int x)
{
     printf("Oranges and lemons, Say the bells of St. Clement’s; ");
     if (x > 1) oranges_and_lemons(x - 1) ;
}.
```

Figure 17: A simple recursive function showing how, via computing, it is possible to create repeated (redundant) copies of ‘Oranges and lemons, Say the bells of St. Clement’s; ’.

Taking things step by step, this works as follows:

1. The pattern ‘`oranges_and_lemons(100)`’ is matched with the pattern ‘`void oranges_and_lemons(int x)`’ in the first line of the function.

2. The two instances of ‘`oranges_and_lemons`’ are unified and the value 100 is assigned to the variable x. The assignment may also be understood in terms of the matching and unification of patterns but the details would be a distraction from the main point here.

3. The instruction ‘`printf("Oranges and lemons, Say the bells of St. Clement’s; ");`’ in the function has the effect of printing out “`Oranges and lemons, Say the bells of St. Clement’s; `”.

4. Then if $x > 1$, the instruction ‘`oranges_and_lemons(x - 1)`’ has the effect of calling the function again but this time with 99 as the value of x (because of the instruction $x-1$ in the pattern ‘`oranges_and_lemons(x - 1)`’, meaning that 1 is to be subtracted from the current value of x).

5. Much as with the first call to the function (item 1, above), the pattern ‘`oranges_and_lemons(99)`’ is matched with the pattern ‘`void oranges_and_lemons(int x)`’ in the first line of the function.

6. Much as before, the two instances of ‘`oranges_and_lemons`’ are unified and the value 98 is assigned to the variable x.

7. This cycle continues until the value of x is 0.

Where does compression of information come in? It happens mainly when one copy of ‘`oranges_and_lemons`’ is matched and unified with another copy so that, in effect, two copies are reduced to one.

There is more about recursion in Appendix C.1.4, below.

C.1.2 A simple example of 'decompression by compression'

In the retrieval of compressed information, the chunking-with-codes idea outlined in Section 2.1.2 provides a simple example of decompression by compression:

- *Compression of information.* If, for example, a document contains many instances of the expression "Treaty on the Functioning of the European Union" we may shorten it by giving that expression a relatively short name or code like 'TFEU' and then replacing all but one instances of the long expression with its shorter code. This achieves compression of information because, in effect, multiple instances of "Treaty on the Functioning of the European Union" have been reduced to one via matching and unification.
- *Retrieval of compressed information.* We can reverse the process and thus decompress the document by searching for instances of 'TFEU' and replacing each one with "Treaty on the Functioning of the European Union". But to achieve that result, the search pattern 'TFEU' needs to be matched and unified with each instance of 'TFEU' in the document. And that process of matching and unification is itself a process of compressing information. Hence, decompression of information has been achieved via IC!

C.1.3 How the SP System may achieve decompression by compression

How the SP System may, with appropriate input, achieve decompression by compression is described in [105, Section 3.8] and [107, Section 4.5]. There are two key points: 1) decompression of a body of information **I**, may be achieved by a process which is *exactly* the same as the process that achieved the original compression of **I**—there is no modification to the program of any kind; 2) all that is needed to achieve decompression is to ensure that there is some residual redundancy in the compressed version of **I**, so that the program has something to work on.

Figure 18 shows a simple example. Here, the SP-multiple-alignment shown in Figure 18 (a), the very simple sentence '`j o h n r u n s`', in row 0 of the SP-multiple-alignment, has been recognised as a sentence comprising a noun followed by a verb.

A 'code' for this analysis may be obtained by scanning the SP-multiple-alignment from left to right, picking out the SP-symbols that have *not* been aligned with any other symbol ([107, Section 4.1], [105, Section 3.5]). The

```
0           j o h n         r u n s       0
            | | | |         | | | |
1      N n1 j o h n #N      | | | |       1
       |            |       | | | |
2 S s0 N            #N V    | | | | #V #S 2
                       |    | | | | |
3                      V v0 r u n s #V    3

(a)
```

```
0 S s0   n1              v0            #S 0
  | |    |               |             |
1 S s0 N |          #N V |          #V #S 1
       | |          |  | |          |
2      N n1 j o h n #N | |          |     2
                       | |          |
3                      V v0 r u n s #V    3

(b)
```

Figure 18: Two SP-multiple-alignments, produced by the SP Computer Model, showing how the program may achieve decompression of information as well as compression of information, as described in the text.

result in this case is the SP-pattern ‘`S s0 n1 v0 #S`’. Without worrying about the details of how many bits are required for each SP-symbol (which has nothing to do with the textual size of each SP-symbol—see [107, Section 4.1] and [105, Section 3.5.2.1]), we can see that there has been a moderate compression of information because 8 SP-symbols in the sentence have been encoded with 5 other SP-symbols.

In Figure 18 (b), the process is reversed. Now the code SP-pattern ‘`S s0 n1 v0 #S`’ is supplied to the program as a New SP-pattern. Each of the SP-symbols in that SP-pattern are given extra bits of information to ensure that the program has some redundancy to work on, as mentioned above. The best SP-multiple-alignment that is created in this case contains ‘`j o h n`’ followed by ‘`r u n s`’, which is of course the original sentence, recreated via its code SP-symbols.

In general, the SP Computer Model, which is devoted to the compression of information, can reverse the process without any modification. It achieves ‘decompression by compression’ without any paradox or contradiction.

C.1.4 How the SP System may create redundancy via recursion

The SP Computer Model may also create redundancy via recursion, as illustrated in Figure 19.

```
0                     0                        0
                      |
1                 X a 0 #X                     1
                  |     |
2             X b X     #X 1 #X                2
              |              |
3         X b X              #X 1 #X           3
          |                       |
4     X b X                       #X 1 #X      4
      |                                |
5 X b X                                #X 1 #X 5
```

Figure 19: One of many SP-multiple-alignments produced by the SP Computer Model with a New SP-pattern, '`0`', and a repository of user-supplied Old SP-patterns: '`X b X #X 1 #X`'. Reproduced with permission from Figure 4.4 (a) in [105].

In this example, the SP Computer Model is supplied with two Old SP-patterns—'`X b X #X 1 #X`' and '`X a 0 #X`'—and a one-symbol New SP-pattern: '`0`'. The program processes this information like this:

1. The SP-symbol '`0`' in the New SP-pattern is matched with, and implicitly unified with, the same SP-symbol in the Old SP-pattern '`X a 0 #X`', as shown in rows 0 and 1 in the figure.

2. The SP-symbols '`X`' and '`#X`' at the beginning and end of '`X a 0 #X`' are matched and unified with the same two symbols at the third and fourth positions in the SP-pattern '`X b X #X 1 #X`', as shown in rows 1 and 2 in the figure.

3. The SP-symbols '`X`' and '`#X`' at the beginning and end of '`X b X #X 1 #X`' are matched and unified with the same two symbols at the third and fourth positions in that same SP-pattern, as shown in rows 2 and 3 in the figure.

4. After that, the process in step 3 repeats, as shown in rows 3 and 4 and rows 4 and 5 of the figure—and it may carry on like this, producing many SP-multiple-alignments, until the operator stops it, or computer memory is exhausted.

If the matching symbols in Figure 19 are all unified (merging each matching pair into a single symbol), the result is a single sequence like this: '`X b X b X b X b X a 0 #X 1 #X 1 #X 1 #X 1 #X`', and likewise for all the many other SP-multiple-alignments that the program may produce. With all but the simplest of those SP-multiple-alignments, there would be redundancy in the repetition of the symbol '`1`', and likewise for other symbols in the figure. Hence, the SP Computer Model has created redundancy by a process which is devoted to the compression of information.

C.2 Redundancy is often useful in the detection and correction of errors and in the storage and processing of information

The fact that redundancy—repetition of information—is often useful in the detection and correction of errors and in the storage and processing of information, and the fact that these things are true in biological systems as well as artificial systems, is the second apparent contradiction to ICHLPC and the SP Theory as a theory of HLPC. Here are some examples:

- *Backup copies.* With any kind of database, it is normal practice to maintain one or more backup copies as a safeguard against catastrophic loss of the data. Each backup copy represents redundancy in the system.

- *Mirror copies.* With information on the internet, it is common practice to maintain two or more mirror copies in different places to minimise transmission times and to spread processing loads across two or more sites, thus reducing the chance of overload at any one site. Again, each mirror copy represents redundancy in the system.

- *Redundancies as an aid to the correction of errors.* Redundancies in natural language can be a very useful aid to the comprehension of speech in noisy conditions.

- *Redundancies in electronic messages.* It is normal practice to add redundancies to electronic messages, in the form of additional bits of information together with checksums, and also by repeating the transmission of any part of a message that has become corrupted. These things help to safeguard messages against accidental errors caused by such things as birds flying across transmission beams, or electronic noise in the system, and so on.

In information processing systems of any kind, uses of redundancy of the kind just described may co-exist with ICMUP. For example: "... it is entirely possible for a database to be designed to minimise internal redundancies and, at the same time, for redundancies to be used in backup copies or mirror copies of the database ... Paradoxical as it may sound, knowledge can be compressed and redundant at the same time." [105, Section 2.3.7].

As noted in Appendix C.1.3, the SP System, which is dedicated to the compression of information, will not work properly with totally random information containing no redundancy. It needs redundancy in its 'New' data in order to achieve such things as the parsing of natural language, pattern recognition, and grammatical inference. Also, for the correction of errors in any incoming batch of New SP-patterns, it needs a repository of Old patterns that represent patterns of redundancy in a previously-processed body of New information.

Figure 20 shows two SP-multiple-alignments that illustrate error correction by the SP Computer Model. Figure 20 (a) shows, as a reference standard, a parsing of the sentence '`t w o k i t t e n s p l a y`' in row 0 where that New SP-pattern is free of errors. For comparison, Figure 20 (b) shows a parsing in which the New SP-pattern in row 0 contains an error of omission ('`t w o`' is changed to '`t o`'), an error of substitution ('`k i t t e n s`' is changed to '`k i t t e m s`'), and an error of addition ('`p l a y`' is changed to '`p l a x y`'). Despite these three errors, the best SP-multiple-alignment created by the SP Computer Model is what would normally be regarded as correct.

This example illustrates the point, mentioned in Appendix B.2, that the exploitation of redundancy for the correction of errors may on occasion be intimately related to the exploitation of redundancy for the compression of information.

C.3 The human mind as a kluge

As mentioned in Section 22, Gary Marcus has described persuasive evidence that, in many respects, the human mind is a kluge. To illustrate the point, here is a sample of what Marcus says:

> "Our memory is both spectacular and a constant source of disappointment: we recognize photos from our high school year-books decades later—yet find it impossible to remember what we had for breakfast yesterday. Our memory is also prone to distortion, conflation, and simple failure. We can know a word but not be

```
0                      t w o              k i t t e n     s                  p l a y           0
                       | | |              | | | | | |     |                  | | | |
1                      | | |         Nr 5 k i t t e n #Nr |                  | | | |           1
                       | | |         |                 |  |                  | | | |
2                      | | |    N Np Nr               #Nr s #N               | | | |           2
                       | | |    | |                         |                | | | |
3               D Dp 4 t w o #D | |                         |                | | | |           3
                |            |  | |                         |                | | | |
4            NP D            #D N |                         #N #NP           | | | |           4
             |                    |                             |            | | | |
5            |                    |                             |       Vr 1 p l a y #Vr       5
             |                    |                             |       |             |
6            |                    |                             |  V Vp Vr           #Vr #V    6
             |                    |                             |  | |                   |
7 S Num    ; NP                   |                           #NP V |                   #V #S 7
     |     |                      |                                 |
8   Num PL ;                      Np                                Vp                        8
```

(a)

```
0                      t   o              k i t t e       m s                  p l a x y           0
                       |   |              | | | | |         |                  | | |   |
1                      |   |         Nr 5 k i t t e n #Nr   |                  | | |   |           1
                       |   |         |                 |    |                  | | |   |
2                      |   |    N Np Nr               #Nr   s #N               | | |   |           2
                       |   |    | |                           |                | | |   |
3               D Dp 4 t w o #D | |                           |                | | |   |           3
                |            |  | |                           |                | | |   |
4            NP D            #D N |                           #N #NP           | | |   |           4
             |                    |                               |            | | |   |
5            |                    |                               |       Vr 1 p l a   y #Vr       5
             |                    |                               |       |               |
6            |                    |                               |  V Vp Vr             #Vr #V    6
             |                    |                               |  | |                     |
7 S Num    ; NP                   |                             #NP V |                     #V #S 7
     |     |                      |                                   |
8   Num PL ;                      Np                                  Vp                          8
```

(b)

Figure 20: (a) The best SP-multiple-alignment created by the SP model with a store of Old SP-patterns like those in rows 1 to 8, representing grammatical structures, including words, and a New SP-pattern in row 0, representing a sentence to be parsed. (b) As in (a) but with errors of omission, commission and substitution in the New SP-pattern, and with same set of Old SP-patterns as before. Figures (a) and (b) are adapted from Figures 1 and 2 in [106], with permission.

> able to remember it when we need it ... or we can learn something valuable ... and promptly forget it. The average high school student spends four years memorising dates, names, and places, drill after drill, and yet a significant number of teenagers can't even identify the *century* in which World War I took place." [54, p. 18, emphasis as in the original].

Clearly, human memory is, in some respects, much less effective than a computer disk drive or even a book. And it seems likely that at least part of the reason for this and other shortcomings of the human mind is that "Evolution [by natural selection] tends to work with what is already in place, making modifications rather than starting from scratch." and "piling new systems on top of old ones" [54, p. 12].

The evidence that Marcus presents is persuasive: it is difficult to deny that, in certain respects, the human mind is a kluge. And evolution by natural selection provides a plausible explanation for anomalies and inconsistencies in the workings of the human mind.

Broadly in keeping with these ideas, Marvin Minsky has suggested that "each [human] mind is made of many smaller processes" called *agents* each one of which "can only do some simple thing that needs no mind or thought at all. Yet when we join these agents in societies—in certain very special ways—this leads to true intelligence." [61, p. 17]. Perhaps errors here and there in a society of agents might explain the anomalies and inconsistencies in human thinking that Marcus has described.

Superficially, evidence and arguments presented by Marcus and Minsky seem to undermine the idea that there is some grand unifying principle—such as IC via SP-multiple-alignment—that governs the organisation and workings of the human mind. But those conclusions are entirely compatible with ICHLPC and the SP Theory as a theory of mind. As Marcus says: "I don't mean to chuck the baby along with its bath—or even to suggest that kluges outnumber more beneficial adaptations. The biologist Leslie Orgel once wrote that 'Mother Nature is smarter than you are,' and most of the time it is." [54, p. 16], although Marcus warns that in comparisons between artificial systems and natural ones, nature does not always come out on top.

In general it seems that, despite the evidence for kluges in the human mind, there can be powerful organising principles too. Since ICHLPC and the SP Theory are well supported by evidence, they are likely to provide useful insights into the nature of human intelligence, alongside an understanding that there are likely to be kluge-related anomalies and inconsistencies too.

Minsky's counsel of despair—"The power of intelligence stems from our vast diversity, not from any single, perfect principle." [61, p. 308]—is probably too strong. It is likely that there is at least one unifying principle for human-level intelligence, and there may be more. And it is likely that, with people, any such principle or principles operates alongside the somewhat haphazard influences of evolution by natural selection.

References

[1] Multiple sequence alignment. In P. Bawono, M. Dijkstra, W. Pirovano, A. Feenstra, S. Abeln, and J. Heringa, editors, *Bioinformatics. Methods in Molecular Biology*, volume 1525, pages 167–189. Humana Press, New York, 2016.

[2] A. Alamia, O. Solopchuk, A. D. Ausilio, V. Van Bever, L. Fadiga, E. Olivier, and A. Zénon. Disruption of Broca's area alters higher-order chunking processing during perceptual sequence learning. *Journal of Cognitive Neuroscience*, 28(3):402–417, 2016.

[3] L. Allison and C. S. Wallace. The posterior probability distribution of alignments and its application to parameter estimation of evolutionary trees and to optimization of multiple alignments. *Journal of Molecular Evolution*, 39:418–430, 1994.

[4] J. J. Atick. Could information theory provide an ecological theory of sensory processing? *Network: Computation in Neural Systems*, 22(1-4):4–44, 2011.

[5] J. J. Atick and A. N. Redlich. What does the retina know about natural scenes? *Neural Computation*, 4:196–210, 1992.

[6] F. Attneave. Some informational aspects of visual perception. *Psychological Review*, 61:183–193, 1954.

[7] V. Balasubramanian. Heterogeneity and efficiency in the brain. *Proceedings of the IEEE*, 103(8):1346–1358, 2015.

[8] H. B. Barlow. Sensory mechanisms, the reduction of redundancy, and intelligence. In HMSO, editor, *The Mechanisation of Thought Processes*, pages 535–559. Her Majesty's Stationery Office, London, 1959.

[9] H. B. Barlow. Trigger features, adaptation and economy of impulses. In K. N. Leibovic, editor, *Information Processes in the Nervous System*, pages 209–230. Springer, New York, 1969.

[10] H. B. Barlow. Intelligence, guesswork, language. *Nature*, 304:207–209, 1983.

[11] H. B. Barlow. Redundancy reduction revisited. *Network: Computation in Neural Systems*, 12:241–253, 2001.

[12] J. D. Barrow. *Pi in the Sky*. Penguin Books, Harmondsworth, 1992.

[13] J. L. Bermúdez. *Thinking Without Words*. Oxford University Press, Oxford, 2003.

[14] W. Bialek, R. R. De Ruyter Van Steveninck, and N. Tishby. Efficient representation as a design principle for neural coding and computation. In *Proceedings of the 2006 IEEE International Symposium on Information Theory*, 2006.

[15] G. Boole. *An Investigation of the Laws of Thought*. Walton and Maberly, London, Kindle edition, 1854.

[16] N. Brenner, W. Bialek, and R. de Ruyter van Steveninck. Adaptive rescaling maximizes information transmission. *Neuron*, 26(3):695–702, 2000.

[17] C. Brown. *My Left Foot*. Vintage Digital, London, Kindle edition, 2014. First published in 1954.

[18] N. Chater. Reconciling simplicity and likelihood principles in perceptual organisation. *Psychological Review*, 103(3):566–581, 1996.

[19] N. Chater. The search for simplicity: a fundamental cognitive principle? *Quarterly Journal of Experimental Psychology*, 52 A(2):273–302, 1999.

[20] N. Chater and M. Oaksford. *The Probabilistic Mind: Prospects for Bayesian Cognitive Science*. Oxford University Press, Oxford, 2008.

[21] N. Chater and P. Vitányi. Simplicity: a unifying principle in cognitive science? *TRENDS in Cognitive Sciences*, 7(1):19–22, 2003.

[22] N. Chater and P. Vitányi. 'Ideal learning' of natural language: positive results about learning from positive evidence. *Journal of Mathematical Psychology*, 51(3):135–163, 2007.

[23] M. Chekaf, N. Cowan, and F. Mathy. Chunk formation in immediate memory and how it relates to data compression. *Cognition*, 155:96–107, 2016.

[24] T. Chilimbi, Y. Suzue, J. Apacible, and K. Kalyanaraman. Project Adam: building an efficient and scalable deep learning training system. In *Proceedings of the 11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 2014)*, pages 571–582. USENIX Association, 2014.

[25] N. Chomsky. *Syntactic Structures.* Mouton, The Hague, 1957.

[26] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* John Wiley & Sons, Hoboken NJ, Second, Kindle edition, 2006.

[27] D. C. Donderi. Visual complexity: a review. *Psychological Bulletin*, 132(1):73–97, 2006.

[28] A. L. Fairhall, G. D. Lewen, W. Bialek, and R. R. de Ruyter van Steveninck. Efficiency and ambiguity in an adaptive neural code. *Nature*, 412:787–792, 2001.

[29] R. Falk and C. Konold. Making sense of randomness: implicit encoding as a basis for judgment. *Psychological Review*, 104(2):301, 1997.

[30] J. Feldman. Minimization of Boolean complexity in human concept learning. *Nature*, 407(6804):630–633, 2000.

[31] J. P. Frisby and J. V. Stone. *Seeing: The Computational Approach to Biological Vision.* The MIT Press, London, England, 2010.

[32] H. G. Furth. *Thinking Without Language: Psychological Implications of Deafness.* The Free Press, 1966.

[33] N. Gauvrit, H. Singmann, F. Soler-Toscano, and H. Zenil. Algorithmic complexity for psychology: a user-friendly implementation of the coding theorem method. *Behavior Research Methods*, 48(1):314–329, 2016.

[34] N. Gauvrit, F. Soler-Toscano, and H. Zenil. Natural scene statistics mediate the perception of image complexity. *Visual Cognition*, 22(8):1084–1091, 2014.

[35] F. Gobet, F. Gobet, P. C. R. Lane, S. Croker, P. C-H. Cheng, G. Jones, I. Oliver, and J. M. Pine. Chunking mechanisms in human learning. *Trends in Cognitive Sciences*, 5(6):236–243, 2001.

[36] M. Gold. Language identification in the limit. *Information and Control*, 10:447–474, 1967.

[37] D. O. Hebb. *The Organization of Behaviour.* John Wiley & Sons, New York, 1949.

[38] F. J. H. Heras, J. Anderson, S. B. Laughlin, and J. E. Niven. Voltage-dependent k^+ channels improve the energy efficiency of signalling in blowfly photoreceptors. *Journal of the Royal Society Interface*, 14:1–13, 2017.

[39] A. M. Hermundstad, J. J. Briguglio, M. M. Conte, J. D. Victor, V. Balasubramanian, and G. Tkaçik. Variance predicts salience in central sensory processing. *eLife*, 3:e03722, 2014.

[40] J. Hernández-Orallo and N. Minaya-Collado. A formal definition of intelligence based on an intensional variant of algorithmic complexity. In *Proceedings of the International Symposium of Engineering of Intelligent Systems (EIS '98)*, pages 146–163, 1998.

[41] A. S. Hsu, N. Chater, and P. Vitáyi. Language learning from positive evidence, reconsidered: a simplicity-based approach. *Topics in Cognitive Science*, 5:35–55, 2013.

[42] M. Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability.* Springer, Berlin, 2005. ISBN 3-540-22139-5, www.hutter1.net/ai/uaibook.htm.

[43] M. Hutter. One decade of universal artificial intelligence. In P. Wang and B. Goertzel, editors, *Theoretical Foundations of Artificial General Intelligence*, volume 4, pages 67–88. Springer, Heidelberg, 2012.

[44] M. Iklé, A. Franz, R. Rzepka, and B. Goertzel. Preface. In M. Iklé, A. Franz, R. Rzepka, and B. Goertzel, editors, *Proceedings of the 11th International Conference, AGI 2018, Prague, Czech Republic, August 22-25, 2018*, volume 10999 of *Lecture Notes in Computer Science*, pages Locations 43–72, Heidelberg, 2018. Springer.

[45] W. Isaacson. *Einstein: His Life and Universe.* Pocket Books, London, Kindle edition, 2007.

[46] B. Julesz. *Foundations of Cyclopean Perception.* Chicago University Press, Chicago, 1971.

[47] S. Kirby, T. Griffiths, and K. Smith. Iterated learning and the evolution of language. *Current Opinion in Neurobiology*, 28:108–114, 2014.

[48] K. Koch, J. McLean, R. Segev, M. A. Freed, M. J. Berry II, V. Balasubramanian, and P. Sterling. How much the eye tells the brain. *Current Biology*, 16(14):1428–1434, 2006.

[49] K. Lamberts and D. Shanks. *Knowledge Concepts and Categories.* Psycholoy Press, Hove, 2013.

[50] S. B. Laughlin1 and T. J. Sejnowski. Communication in neuronal networks. *Science*, 301(5641):1870–1874, 2003.

[51] B. Lemaire, V. Robinet, and S. Portrat. Compression mechanisms in working memory. *Mathématiques et Sciences Humaines*, 199(3):71–84, 2012.

[52] M. Li and P. Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications.* Springer, New York, 3rd edition, 2014.

[53] B. Z. Mahon and A. Caramazza. Concepts and categories: A cognitive neuropsychological perspective. *Annual Review of Psychology*, 60:27–51, 2009.

[54] G. Marcus. *Kluge: the Hapharzard Construction of the Human Mind.* Faber and Faber, London, paperback edition, 2008. ISBN: 978-0-571-23652-7.

[55] D. Marr. *Vision: a Computational Investigation into the Human Representation and Processing of Visual Information.* W.H. Freeman, San Francisco, 1982.

[56] D. Marr and T. Poggio. A computational theory of human stereo vision. *Proceedings of the Royal Society of London. Series B*, 204(1156):301–328, 1979.

[57] S. Martinez-Conde, J. Otero-Millan, and S. L. Macknik. The impact of microsaccades on vision: towards a unified theory of saccadic function. *Nature Reviews Neuroscience*, 14:83–96, 2013.

[58] F. Mathy and J. Feldman. What's magic about magic numbers? Chunking and data compression in short-term memory. *Cognition*, 122(3):346–362, 2012.

[59] C. B. Mervis and E. Rosch. Categorization of natural objects. *Annual Review of Psychology*, 32:89–115, 1981.

[60] G. A. Miller. The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review*, 63:81–97, 1956.

[61] M. Minsky, editor. *The Society of Mind.* Simon & Schuster, New York, 1986.

[62] A. Newell. You can't play 20 questions with nature and win: Projective comments on the papers in this symposium. In W. G. Chase, editor, *Visual Information Processing*, pages 283–308. Academic Press, New York, 1973.

[63] A. Newell, editor. *Unified Theories of Cognition.* Harvard University Press, Cambridge, Mass., 1990.

[64] I. Newton. *The Mathematical Principles of Natural Philosophy.* The Perfect Library, Kindle edition, 2014. First published 1687. Illustrated and bundled with *Life of Sir Isaac Newton.*

[65] B. A. Olshausen and D. J. Field. Vision and the coding of natural images: the human brain may hold the secrets to the best image-compression algorithms. *American Scientist*, 88(3):238–245, 2000.

[66] B. A. Olshausen and D. J. Field. Sparse coding of sensory inputs. *Current Opinion in Neurobiology*, 14(4):481–487, 2004.

[67] S. E. Palmer, O. Marre, M. J. Berry II, and W. Bialek. Predictive information in a sensory population. *PNAS*, 112(22):6908–6913, 2015.

[68] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann, San Francisco, revised second printing edition, 1997.

[69] E. M. Pothos and N. Chater. A simplicity principle in unsupervised human categorization. *Cognitive Science*, 26:303–343, 2002.

[70] F. Ratliff, H. K. Hartline, and W. H. Miller. Spatial and temporal aspects of retinal inhibitory interaction. *Journal of the Optical Society of America*, 53:110–120, 1963.

[71] J. Rissanen. Modelling by the shortest data description. *Automatica*, 14(5):465–471, 1978.

[72] J. Rissanen. Stochastic complexity. *Journal of the Royal Statistical Society B*, 49(3):223–239, 1987.

[73] V. Robinet, B. Lemaire, and M. B. Gordon. MDLChunker: a MDL-based cognitive codel of inductive learning. *Cognitive Science*, 35:1352–1389, 2011.

[74] K. Sakai, K. Kitaguchi, and O. Hikosaka. Chunking during human visuomotor sequence learning. *Experimental Brain Research*, 152:229–242, 2003.

[75] K. Sayood. *Introduction to Data Compression.* Morgan Kaufmann, Amsterdam, 2012.

[76] J. Schmidhuber. Deep learning in neural networks: an overview. *Neural Networks*, 61:85–117, 2015.

[77] B. Sengupta, S. B. Laughlin, and J.y E. Niven. Consequences of converting graded to action potentials upon neural information coding and energy efficiency. *PLOS Computational Biology*, 10(1):e1003439, 2014.

[78] C. E. Shannon and W. Weaver. *The Mathematical Theory of Communication.* University of Illinois Press, Urbana, 1949.

[79] D. J. Simons and C. F. Chabris. Gorillas in our midst: sustained inattentional blindness for dynamic events. *Perception*, 28(9):1059–1074, 1999.

[80] D. J. Simons and D. T. Levin. Failure to detect changes to people during a real-world interaction. *Psychonomic Bulletin & Review*, 5(4):644–649, 1998.

[81] R. R. Sokal and P. H. A. Sneath, editors. *Numerical Taxonomy: the Principles and Practice of Numerical Classification.* W. H. Freeman, San Francisco, 1973.

[82] R. J. Solomonoff. A formal theory of inductive inference. Parts I and II. *Information and Control*, 7:1–22 and 224–254, 1964.

[83] R. J. Solomonoff. The discovery of algorithmic probability. *Journal of Computer and System Sciences*, 55(1):73–88, 1997.

[84] L. R. Squire, D. Berg, F. E. Bloom, S. du Lac, A. Ghosh, and N. C. Spitzer, editors. *Fundamental Neuroscience.* Elsevier, Amsterdam, fourth edition, 2013.

[85] M. Tamariz and S. Kirby. Culture: copying, compression and conventionality. *Cognitive Science*, 39(1):171–183, 2015.

[86] W. J. Teahan and K. M. Alhawiti. Preprocessing for PPM: compressing UTF-8 encoded natural language text. *International Journal of Computer Science & Information Technology*, 7(2):41–51, 2015.

[87] T. Teşileanu, B. Ölveczky, and V. Balasubramanian. Rules and mechanisms for efficient two-stage learning in neural circuits. *eLife*, 6:e20944, 2017.

[88] W. Thomson. *Outline of the Laws of Thought.* William Pickering, London, 1842. Republished in the Leopold Classic Library.

[89] A. M. Turing. Intelligent machinery. In B. J. Copeland, editor, *The Essential Turing: Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life: Plus The Secrets of Enigma*, pages 395–432. Oxford University Press, Oxford, 1948.

[90] J. Veness, K. S. Ng, M. Hutter, W. Uther, and D. Silver. A Monte Carlo AIXI approximation. *Journal of Artificial Intelligence Research*, 40(1):95–142, 2011.

[91] P. Vitányi and N. Chater. Identification of probabilities. *Journal of Mathematical Psychology*, 76(Part A):13–24, 2017.

[92] G. von Békésy. *Sensory Inhibition.* Princeton University Press, Princeton, NJ, 1967.

[93] C. S. Wallace and D. M. Boulton. An information measure for classification. *Computer Journal*, 11(2):185–195, 1968.

[94] C. S. Wallace and P. R. Freeman. Estimation and inference by compact coding. *Journal of the Royal Statistical Society B*, 49(3):240–252, 1987.

[95] S. Watamabe. Information-theoretical aspects of inductive and deductive inference. *IBM Journal of Research and Development*, 4:208–231, 1960.

[96] S. Watanabe, editor. *Frontiers of Pattern Recognition.* Academic Press, New York, 1972.

[97] S. Watanabe. Pattern recognition as information compression. In *Frontiers of Pattern Recognition* [96].

[98] C. S. Webster. Alan turing's unorganized machines and artificial neural networks: his remarkable early work and future possibilities. *Evolutionary Intelligence*, 5:35–43, 2012.

[99] J. G. Wolff. An algorithm for the segmentation of an artificial language analogue. *British Journal of Psychology*, 66:79–90, 1975. See www.cognitionresearch.org/lang_learn.html#wolff_1975.

[100] J. G. Wolff. The discovery of segments in natural language. *British Journal of Psychology*, 68:97–106, 1977. bit.ly/Yg3qQb.

[101] J. G. Wolff. Language acquisition and the discovery of phrase structure. *Language & Speech*, 23:255–269, 1980. See www.cognitionresearch.org/lang_learn.html#wolff_1980.

[102] J. G. Wolff. Language acquisition, data compression and generalization. *Language & Communication*, 2:57–89, 1982. doi.org/10.1016/0271-5309(82)90035-0, bit.ly/Zq0zAl.

[103] J. G. Wolff. Learning syntax and meanings through optimization and distributional analysis. In Y. Levy, I. M. Schlesinger, and M. D. S. Braine, editors, *Categories and Processes in Language Acquisition*, pages 179–215. Lawrence Erlbaum, Hillsdale, NJ, 1988. bit.ly/ZIGjyc.

[104] J. G. Wolff. Computing, cognition and information compression. *AI Communications*, 6(2):107–127, 1993. bit.ly/XL359b.

[105] J. G. Wolff. *Unifying Computing and Cognition: the SP Theory and Its Applications*. CognitionResearch.org, Menai Bridge, 2006. ISBNs: 0-9550726-0-3 (ebook edition), 0-9550726-1-1 (print edition). Distributors, including Amazon.com, are detailed on bit.ly/WmB1rs.

[106] J. G. Wolff. Towards an intelligent database system founded on the SP theory of computing and cognition. *Data & Knowledge Engineering*, 60:596–624, 2007. arXiv:cs/0311031 [cs.DB], bit.ly/1CUldR6.

[107] J. G. Wolff. The SP Theory of Intelligence: an overview. *Information*, 4(3):283–341, 2013. arXiv:1306.3888 [cs.AI], bit.ly/1NOMJ6l.

[108] J. G. Wolff. Application of the SP Theory of Intelligence to the understanding of natural vision and the development of computer vision. *SpringerPlus*, 3(1):552–570, 2014. arXiv:1303.2071 [cs.CV], bit.ly/2oIpZB6.

[109] J. G. Wolff. Information compression, multiple alignment, and the representation and processing of knowledge in the brain. *Frontiers in Psychology*, 7:1584, 2016. arXiv:1604.05535 [cs.AI], bit.ly/2esmYyt.

[110] J. G. Wolff. The SP Theory of Intelligence: its distinctive features and advantages. *IEEE Access*, 4:216–246, 2016. arXiv:1508.04087 [cs.AI], bit.ly/2qgq5QF.

[111] J. G. Wolff. Software engineering and the SP Theory of Intelligence. Technical report, CognitionResearch.org, 2017. Submitted for publication, arXiv:1708.06665 [cs.SE], bit.ly/2w99Wzq.

[112] J. G. Wolff. Introduction to the SP Theory of Intelligence. Technical report, CognitionResearch.org, 2018. arXiv:1802.09924, bit.ly/2ELq0Jq.

[113] J. G. Wolff. Mathematics as information compression via the matching and unification of patterns. Technical report, CognitionResearch.org, 2018. Submitted for publication. arXiv:1808.07004 [cs.AI], bit.ly/2LWbjtK.

[114] J. G. Wolff. Solutions to problems with deep learning. Technical report, CognitionResearch.org, 2018. arXiv:1801.05457 [cs.LG], bit.ly/2AJzu4j.

[115] M. Yamaguchi and G. D. Logan. Pushing typists back on the learning curve: Revealing chunking in skilled typewriting. *Journal of Experimental Psychology: Human Perception and Performance*, 40(0):592–612, 2014.