

A Roadmap for the Development of the ‘SP Machine’ for Artificial Intelligence

VASILE PALADE¹ AND J. GERARD WOLFF^{2*}

¹Coventry University, Priory Street, Coventry, United Kingdom, CV1 5FB, UK

²CognitionResearch.org, 18 Penlon, Menai Bridge, Anglesey, LL59 5LR, UK

*Corresponding author: jgw@cognitionresearch.org

This paper describes a roadmap for the development of the *SP Machine*, based on the *SP Theory of Intelligence* and its realization in the *SP Computer Model*. The *SP Machine* will be developed initially as a software virtual machine with high levels of parallel processing, hosted on a high-performance computer. The system should help users visualize knowledge structures and processing. Research is needed into how the system may discover low-level features in speech and in images. Strengths of the *SP System* in the processing of natural language may be augmented, in conjunction with the further development of the *SP System*’s strengths in unsupervised learning. Strengths of the *SP System* in pattern recognition may be developed for computer vision. Work is needed on the representation of numbers and the performance of arithmetic processes. A computer model is needed of *SP-Neural*, the version of the *SP Theory* expressed in terms of neurons and their interconnections. The *SP Machine* has potential in many areas of application, several of which may be realized on short-to-medium timescales.

Keywords: information compression; artificial intelligence; natural language processing; pattern recognition; computer vision; neuroscience

Received 2 February 2018; revised 3 August 2018; editorial decision 25 October 2018

Handling editor: Fionn Murtagh

1. INTRODUCTION

The main purpose of this paper is to describe a roadmap for the development of an industrial-strength *SP Machine*, a ‘new generation’ computing system with intelligence based on the *SP Theory of Intelligence* and its realization in the *SP Computer Model*, two things which, together, may be referred to as the *SP System*. That system, outlined briefly in Section 1.1 and more fully in Appendix A is the product of a long-term programme of research by the second author, begun in 1987 with a break between early 2006 and late 2012.

This paper is radically different from papers that report the procedures and results of one or more experiments that have been conducted already, or papers that review a range of such reports in a given field. Instead, this paper provides a roadmap for future developments, looking forward to a whole programme of research to develop the *SP Machine*.

Early ideas about the *SP System* were described in this journal in [1]. Relatively full descriptions of the *SP System* as it is now may be found in the book *Unifying Computing and*

Cognition [2] and the paper ‘The *SP Theory of Intelligence*: an overview’ [3]. Details of several other papers, with download links, may be found on www.cognitionresearch.org/sp.htm.

At all but the earliest stages, development and testing of the *SP Computer Model* have been a valuable aid to theorizing: helping to reduce vagueness in the theory, as a means of testing whether ideas work as anticipated, and as a means of demonstrating what the system can do. It is envisaged that, as the *SP Machine* is developed, there will be similar benefits yielding further insights and further development of the *SP Theory*.

It is anticipated that, when it is mature, there will be many applications for the *SP Machine*. Some of these potential applications are described in Appendix A.8.

1.1. The *SP Theory* and Computer Model

The SP Theory of Intelligence, with its realization in the SP Computer Model, is a unique attempt to simplify and integrate observations and concepts across artificial intelligence,

mainstream computing, mathematics and human learning, perception and cognition, with information compression as a unifying theme. It is described in Appendix A with enough detail to allow the rest of the paper to be understood.

The SP System has several distinctive features and advantages compared with AI-related alternatives [4].

The most distinctive feature of the SP System is the powerful concept of *SP-multiple-alignment*, borrowed and adapted from the concept of ‘multiple sequence alignment’ in bioinformatics. A key advantage of the SP-multiple-alignment framework as it has been developed in the SP System is its versatility in aspects of intelligence, its versatility 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. Seamless integration of diverse aspects of intelligence and diverse kinds of knowledge appears to be *essential* in any artificial system that aspires to the fluidity, versatility, and adaptability of human intelligence.

1.2. The SP Machine

It is anticipated that a fully-developed SP Machine will provide a means of realizing the several potential benefits and applications of the SP System, as shown schematically in Fig. 1. Some of these potential benefits and applications are described in [5] and in the several papers referenced in Appendix A.8.

It is envisaged that, initially, the SP Machine will be developed as a highly parallel software virtual machine, hosted on an existing high-performance computer, and with a user interface that helps users to visualize knowledge structures in the system and how they are processed.

Although the SP Computer Model is still useful as a vehicle for research (Section 6), the main function of the SP Machine as just described would be to open things up to facilitate research across a wider range of issues. It may also prove useful as a means of demonstrating what the SP System can do.

At some stage, there is likely to be a case for the development of new hardware, dedicated to structures and processes in the SP System, and with potential for substantial gains in efficiency and performance ([6], [7, Section IX]).

1.3. Presentation

After a summary of the research strategy which is envisaged (Section 3), several sections describe things that need to be done in the development of the SP Machine, including a description of problems to be solved, and potential solutions to those problems, including some areas of application that may yield benefits on relatively short timescales.

Four of the sections are about pieces of ‘unfinished business’ in the development of the SP Machine, outlined in [3, Section 3.3]: processing information in two or more dimensions

(Section 9); recognition of perceptual features in speech and visual images (Section 10); unsupervised learning (Section 12); and processing numbers (Section 14).

1.4. An invitation

In accordance with the saying that ‘A camel is a horse designed by a committee’, the development of a scientific theory, like writing a novel or composing a symphony, is a task that is not easily shared. But now that the SP Theory is relatively mature, it should become easier progressively for other researchers to work on the several avenues that now need to be explored, without treading on each others’ toes.

The involvement of other researchers would be very much welcomed. The authors of this paper would be happy to discuss possibilities, and the necessary facilities.

2. BACKGROUND

The *SP Theory of Intelligence* and its realization in the *SP Computer Model* is the product of an extended programme of research by the second author.

In accordance with widely accepted principles in science, the overarching goal of this programme of research has been the simplification and integration of observations and concepts across artificial intelligence, mainstream computing, mathematics, and human learning, perception, and cognition, with information compression as a unifying theme.

In this respect, the research contrasts sharply with the widespread and longstanding tendency for research in AI to become fragmented into many subfields, described very well by Pamela McCorduck [8].¹ The research is much more in the spirit of exhortations—by Allen Newell [9, 10], by researchers in ‘artificial general intelligence’ (e.g. [11]), [12] and others)—for researchers in AI and related areas to adopt a high-level global view of their field, and to seek unifying principles across wide areas.

The SP Theory draws on an earlier programme of research (summarized in [13]) developing computer models of how a child learns his or her first language or languages. The emphasis on hierarchical structures in those models, excluding other kinds of structure, proved unsuitable for the SP programme of research because of the much wider scope of the later research. But information compression remains a key principle in the SP research as it proved to be in the earlier research.

After several years of work by the second author—reading, thinking and experimentation with computer models—it gradually became clear that a modified version of ‘multiple

¹The goals once articulated with debonair intellectual verve by AI pioneers appeared unreachable ... Subfields broke off—vision, robotics, natural language processing, machine learning, decision theory—to pursue singular goals in solitary splendour, without reference to other kinds of intelligent behaviour’ [8, p. 417].

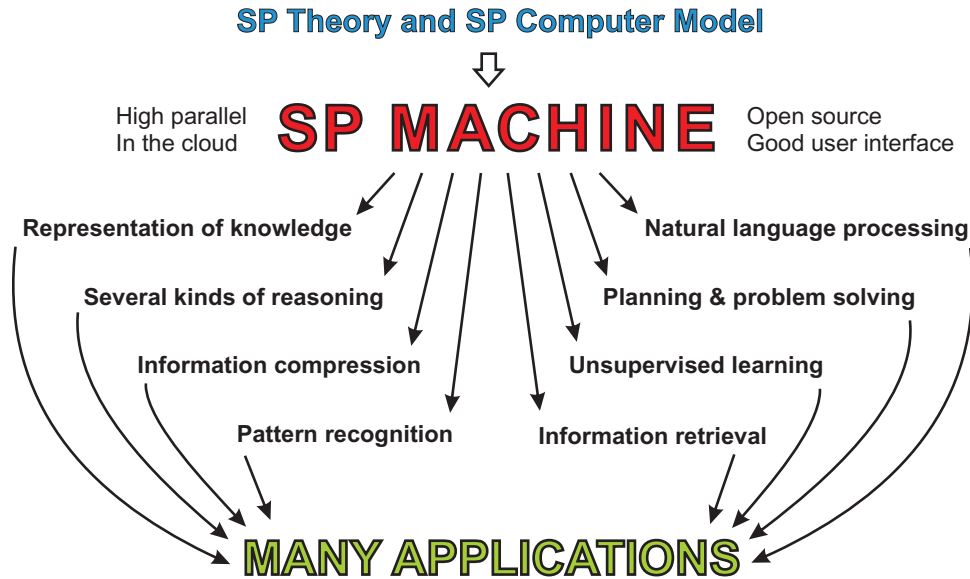


FIGURE 1. A schematic view of how the SP Machine may develop from the SP Theory and the SP Computer Model. Adapted with permission from Fig. 6 in [6].

sequence alignment’ from bioinformatics had the potential to provide a good unifying framework across artificial intelligence, mainstream computing, mathematics, and human learning, perception and cognition, with information compression as a unifying theme.

This new framework—*SP-multiple-alignment*—has been progressively refined to provide a robust computational framework for diverse aspects of intelligence, for the representation of diverse kinds of knowledge, and for their seamless integration in any combination, where ‘any combination’ means a set comprising all the afore-mentioned aspects of intelligence and diverse kinds of knowledge, or any subset of that set.

3. OVERVIEW OF RESEARCH STRATEGY

As already mentioned, the ‘SP’ research, up to now, is a unique attempt to simplify and integrate observations and concepts across a broad canvass, with information compression as a unifying theme. We envisage that the same principles will apply in the future developments described in this paper.

Already, this long-term objective is showing many potential dividends:

- The strengths of SP-multiple-alignment ([2, Chapter 3], [4, Section 4]) in the simplification and integration of observations and concepts across a broad canvass are a major step forward with clear potential in the long-sought-after quest

for general, human-level artificial intelligence and for developing an understanding of human intelligence.

- As described in Appendix A.8, the SP System has many potential benefits and applications.

3.1. Maintain a strong focus on simplicity and power

The long view of this research contrasts with, for example, most research on deep learning in artificial neural networks where undoubted successes obscure several weaknesses in the framework (Appendix A.7), suggesting that deep learning is likely to be a blind alley in any search for general human-level artificial intelligence.

For these kinds of reasons, researchers working with the SP concepts are encouraged to maintain a strong focus on maintaining a favourable combination of conceptual simplicity with descriptive or explanatory power, resisting the temptation to take short cuts or to introduce ideas that do not integrate well with the SP framework. That said, the following kinds of development, which are not mutually exclusive, appear to be reasonable:

- There may be opportunities to develop SP capabilities with industrial strength for practical applications, without corrupting the core concepts (Section 18). An example is the potential of the SP Machine to serve as a database management system with versatility and intelligence [14].

- There may be a case, on occasion, to develop a hybrid system as a stop-gap to facilitate practical applications, pending the development of a solution that is theoretically clean. An example in computer vision would be the use of existing procedures for the detection of low-level features in images (such as lines and corners) pending an analysis in terms of SP concepts (Section 10).

3.2. Scheduling

As described later, some things are probably best done before others. These ‘ground-clearing’ tasks are described early in what follows. When those tasks have been completed, it should become easier to develop two or more aspects of the SP Machine in parallel, in tasks described in later subsections.

When the SP System has been brought close to the point where it may prosper in a commercial environment, perhaps via applications that may be developed on relatively short timescales (Section 18), the system may be refined in specific application areas.

4. THE CREATION OF A HIGHLY PARALLEL SP MACHINE

The proposed development of the SP Machine should begin with SP71, the latest version of the SP Computer Model, outlined in Appendix A.5. Instructions for obtaining the source code are in Appendix B.

To get things going, the first stage, as outlined in this main section, is probably the development of the SP Machine with high levels of parallel processing, and then, in Sections 7 and 8, the creation of a user interface that is easy to use and that facilitates the visualization of knowledge structures and the visualization of the processing of knowledge.

4.1. Is parallel processing really needed?

A substantial advantage of the SP System, compared with, for example, deep learning in artificial neural networks [15], is that, on the strength of evidence to date, the SP Machine is likely to be very much less demanding of computational resources than deep learning systems [4, Section V-E]. And there appears to be potential to speed up processing by the judicious use of indexing, as outlined in Section 5.

More generally, there appears to be potential in the SP System for very substantial gains in computational efficiency compared with conventional computers: by the use of a technique called ‘model-based coding’ for the efficient transmission of data [6, Section VIII]; and by exploiting the statistical information which is gathered as a by-product of how the system works [6, Section IX].

So it is reasonable to ask whether the existing computer model (SP71), or some successor to it, might be speedy enough as it is, without the need for parallel processing? Here are some answers to that question which are not mutually exclusive:

- Although, as mentioned above, the SP Machine has the potential to be relatively frugal in its computational demands, its full potential in that respect will probably not be fully realized until new hardware has been developed to take full advantage of opportunities in the structure and workings of the SP System.
- The afore-mentioned gains in efficiency in the transmission of information via model-based coding, will not be fully realized until unsupervised learning in the SP System is more fully developed.
- Although, with AI problems, the SP System is likely to be much more efficient than deep learning systems, AI problems are normally quite computationally demanding compared with, for example, school-level mathematics. For that kind of reason, parallel processing is likely to be needed to achieve acceptable run times when the SP Machine is applied to what is normally the relatively demanding task of learning in unsupervised mode, and perhaps also to such things as pattern recognition and the processing of natural languages.
- When the SP Machine has been generalized to work with 2D patterns, for reasons described in Section 9.1, it is likely that parallel processing will be needed to speed up processing in what is likely to be the computationally intensive process of finding good full or partial matches between such patterns.
- Parallel processing will almost certainly be needed if the SP Machine is to become an industrial-strength database management system with intelligence, as described in [14].

In short, there are several reasons why the SP Machine should exploit parallel processing.

4.2. How parallel processing may be applied

The effective application of parallel processing will require careful analysis of the SP71 computer model (Appendix A.5) to identify parts of the programme where parallel processing may be applied.

A consistency model, such as Leslie Lamport’s ‘sequential consistency model’ [16], or one of its successors, will be needed to ensure that any parallel version of the SP Computer Model produces the same results as the original model.

There may be a case for using an appropriate calculus or algebra, such as ‘Communicating Sequential Processes’ [17],

to facilitate reasoning about the workings of the parallel system.

At this stage, it seems likely that parallel processing may be applied most effectively via a MapReduce technique (see, for example [18]). It may perhaps be applied in these parts of the programme:

- With *the building of SP-multiple-alignments* (Appendix A.5.2), a recurring operation is searching for good full or partial matches between one ‘driving’ SP-pattern and each of a set of ‘target’ SP-patterns (Appendix A.5.1), where the set of target SP-patterns may become very large. Here, the ‘map’ phase would be applying the search in parallel to each pairing of a driving SP-pattern with a target SP-pattern, while the ‘reduce’ phase would be identifying the most successful matches and selecting a subset for further processing.
- With *unsupervised learning*, each SP-pattern in a relatively large set of New SP-patterns is processed via a search for SP-multiple-alignments that allow the given New SP-pattern to be encoded economically. Here, the ‘map’ phase would be the application of the search process to all the New SP-patterns in parallel, and the ‘reduce’ phase would be a process of sifting and sorting to create one or more alternative grammars for the set of New SP-patterns.

There may be other opportunities for applying the MapReduce technique in the SP Machine. There may also be opportunities for the use of other techniques such as pipelining (see, for example [19]).

4.3. Computing environments for the application of parallel processing

For this development, we have identified two main kinds of computing environment that may be used for the application of parallel processing:

- *A stand-alone computer cluster.* A server with one or more multi-core chips and appropriate software is a promising vehicle for the research. We believe this is likely to be better than most large-scale ‘supercomputing’ facilities, because it would give researchers unfettered access to develop programmes and to test them frequently, without restrictions.
- *Highly parallel search mechanisms in a search engine.* An interesting alternative would be to replace the low-level search mechanisms in the SP Machine with highly parallel search mechanisms in an existing search engine. This would supercharge the SP Machine with the power of the given search engine, and would in

effect add intelligence to the search engine. Here is a little more detail:

- (1) It appears that the most computationally intensive part of the SP Computer Model is the process of finding good full and partial matches between pairs of patterns, as described in [2, Appendix A].
- (2) Finding good full and partial matches between pairs of patterns is one of the things that can be achieved by any of the leading search engines.
- (3) With any of the leading search engines, that kind of searching can be done with remarkable speed.
- (4) Part of this speed is due to indexing, but the application of high levels of parallel processing is at least as important.
- (5) These observations suggest that it makes sense to try to speed up the SP System via the application of highly parallel search mechanisms in a search engine, including the use of indexing (Section 5).

The SP Machine, hosted on a search engine, may approach human-like intelligence with access to all the knowledge that has been indexed in the given search engine. Looking further ahead, all search engines may be adapted in this way, and some of the processing may be offloaded to PCs and other computers around the world, in the manner of volunteer computing projects such as SETI@home (bit.ly/2o37khu) and Folding@home (bit.ly/2AJmXTZ).

Many benefits may flow from artificial intelligence like that, in the Internet and beyond. But there are also potential dangers and other problems which will need careful scrutiny and much debate.

5. INDEXING AND HASH CODING AS POSSIBLE MEANS OF SPEEDING UP PROCESSING

A prominent feature of the SP System, both in the way it does unsupervised learning and the way it builds SP-multiple-alignments, is a process of searching for SP-symbols that match each other, which is normally part of a process of searching for full or partial matches between SP-patterns.

Since that kind of searching for matches between symbols is often done repeatedly with the same sets of New and Old SP-patterns, there is potential for substantial gains in computational efficiency by recording matches as they are found and using that information to by-pass searching in later processing. The record of matches that have been found is, in effect, an index into the data. This may be done with matches between individual SP-symbols, and may perhaps also be done for matches between SP-patterns.

In the process of finding the entry, if any, for any given symbol in the index mentioned above, there may be a case for introducing hash coding as a means of increasing the

speed of that processing, as described in, for example, [20, pp. 61–65].

As outlined in Section 15.2, similar ideas may be applied in *SP-Neural*, a version of the SP Theory expressed in terms of neurons and their interconnections.

6. RESEARCH THAT MAY BE DONE WITH THE SP COMPUTER MODEL

It is envisaged that the highly parallel SP Machine with a ‘friendly’ user interface will be a good foundation for further development of the SP Machine. But pending that development, quite a lot can be done with the SP Computer Model. The main areas that may need to wait for completion of the highly parallel SP Machine are:

- *Processing information in two or more dimensions*, Section 9. This is likely to be relatively demanding for computing power.
- *The processing of natural language*, Section 11. Progress can be made in experimenting with small-scale samples of English-like artificial languages that exhibit structures of interest. But it is likely that significantly more power will be needed for the processing of realistically large bodies of natural language.
- *Unsupervised learning*, Section 12. As with natural language, some progress can be made with small examples. But unsupervised learning with realistically large bodies of data is likely to require the power of the highly parallel SP Machine.
- *Computer vision*, Section 13. As with the processing of 2D SP-patterns, computer vision is likely to be relatively demanding of computer power.
- *SP-Neural*, Section 15. Early work in the development of a computer model of SP-Neural may be done with the SP Computer Model. But more power will probably be needed later.
- *Development of applications of the SP Machine*, Section 17. It will be best to develop most of the SP potential applications on the highly parallel SP Machine.

7. DEVELOPMENT OF A ‘FRIENDLY’ USER INTERFACE

The user interface for the SP71 computer model is the product of *ad hoc* developments in response to evolving needs, without any attempt at a coherent and streamlined design.

This has not mattered for the one user (the second author) who originated each feature and has learned the workings of the user interface as it evolved. However, as the SP System is opened up to new users, a more intuitive and ‘friendly’ user interface is needed, taking advantage of advanced graphical techniques that are available now.

Since all users, both local and remote, may access the SP Machine via a web browser, and to eliminate the need for users to download any software, it is envisaged that the new user interface will be developed using HTML5 or one of its successors.

8. DEVELOPMENT TO FACILITATE VISUALIZATION OF STRUCTURES AND PROCESSES

An important concern with artificial intelligence is that it should be possible to inspect and understand how an AI system stores its knowledge and how it reaches its conclusions. This is important where an AI system may have an impact on the safety of people or where its performance and conclusions may have a bearing on court cases.

A major strength of the SP System is that all its knowledge is open to inspection and it provides an audit trail for all its processing [6, Section XI].

Although this information is provided by the system, there is probably scope for improving its presentation, so that structures and processes can be seen easily and so that users can find easily what they are looking for.

9. PROCESSING INFORMATION IN TWO OR MORE DIMENSIONS

It has been recognized for some time that the SP Machine should be generalized to work with information in two dimensions and probably more [2, Section 13.2.1]. How this may be done is discussed in the following subsections.

9.1. Introduction of 2D SP-patterns

The SP Computer Model as it is now works only with 1D patterns. This is good for modelling such things as natural language texts, grammars for natural or artificial languages, or logical or mathematical expressions, but it is not well suited to the representation and processing of such things as photographs, paintings, drawings and diagrams.

Those last-mentioned kinds of entity provide the main motive for introducing 2D SP-patterns into the SP System. But 2D SP-patterns may also serve in the definition and execution of procedures that operate in parallel [7, Sections V–G], and, as described in Section 9.2, 2D SP-patterns can play a useful part in the representation and processing of 3D structures.

It is a simple enough matter to add 2D SP-patterns to the SP System. What is more challenging is how the concept of SP-multiple-alignment may be generalized to incorporate 2D SP-patterns as well as 1D SP-patterns, how to represent SP-multiple-alignments, which include 2D SP-patterns on a 2D computer screen or a printed page, and how unsupervised

learning may work with 2D SP-patterns as well as 1-D SP-patterns.

Some clues may be gleaned from how, in digital photography, pictures that overlap each other may be stitched together to create a panorama. However, it seems unlikely that software that is designed for the stitching together of overlapping pictures will have the flexibility needed to achieve the 2D equivalent of the processes in SP71 that can find good full and partial matches between two 1D SP-patterns, including cases where there are substantial differences between the SP-patterns (Appendix A.5.1).

Generalizing that process is likely to be a significant challenge. It is likely also that any reasonably good solution will absorb quite a lot of computing power. To achieve useful speeds in processing, it is likely that high levels of parallel processing will be required, as noted in Section 4.1.

9.2. Modelling structures in three dimensions

At first sight, the obvious way to model 3D structures in the SP System is to add 3D SP-patterns to the system. But that is probably a step too far, partly because of the complexity of matching patterns in three dimensions and the difficulty of representing SP-multiple-alignments of 3D SP-patterns, and partly for reasons described here:

‘Patterns in one or two dimensions may serve well enough to represent concepts with three or more dimensions. There is a long tradition of representing buildings and other engineering artifacts with 2D plans and elevations.’ [2, Section 13.2.2].

and

‘... there is a distinct tendency for [the brain] to be organized into layers of cells such as those found in the cortex and in the lateral geniculate body. Although these are often folded into three dimensions, they are topologically equivalent to two-dimensional sheets. ... these sheets of neurons would provide a natural home for SP-patterns.’ *ibid.*

The case for using 2D SP-patterns as a means of building and representing 3D structures is also strengthened by the existence of applications that work like that, such as ‘BigObjectBase’, bit.ly/2V1Koh5 and ‘PhotoModeller’, bit.ly/2rNIPG9, and also Google’s ‘Streetview’,² which builds 3D models of street plans from 2D photographs. There is relevant discussion in [21, Sections 6.1 and 6.2].

9.3. Modelling structures in four or more dimensions

Although there are four dimensions in our everyday experience—three dimensions of space and one of time—it seems likely that, normally, people do not attempt to integrate them

in the manner of Einstein’s space-time but concentrate on a subset of the four dimensions, such as the succession of 2D images on our retinas as we walk through a building or drive around a town. It seems likely that, for most applications, that kind of partial integration would work well enough.

It seems likely that an important application of the SP Machine will be in learning 2D and 3D structures that change across the time dimension—from video recordings in which there will be a succession of 2D frames. With 3D videos it seems likely that preprocessing will be needed to extract structural information from each stereo pair of frames, followed by processing of the time dimension to extract regularities across time.

Extracting regularities across time is something that is likely to play an important part in how the SP System comes to isolate such things as people, birds, cars and so on as discrete entities, and classes of such things, in the system’s conceptual world, and it is likely to be important in encoding such concepts as motion and speed [21, Section 5].

If for any reason, there is a need to represent structures in five or more dimensions, it is probably best to borrow techniques used by mathematicians for dealing with such structures.

9.4. Unordered patterns or sets

As noted in Appendix A.4, there may be a case for introducing unordered SP-patterns or sets, where the order of the SP-symbols is not significant. But that may not be necessary since sets of New SP-patterns have no intrinsic order and that may be sufficient for the modelling of unordered groupings, as described in [22, Section 3.4].

10. RECOGNITION OF LOW-LEVEL PERCEPTUAL FEATURES IN SPEECH AND IMAGES

As noted in Appendix A.4, the SP System is designed to represent and process stored knowledge and sensory data expressed as atomic *SP-symbols* in *SP-patterns*. So at first sight, the SP System should be able to work with any digital information, meaning information that is founded on the atomic symbols ‘0’ and ‘1’. But there are two main complications:

- Much digital information drawn relatively directly from the environment is encoded as, for example, hexadecimal numbers representing such things as levels of brightness in an image. Until the SP System can deal effectively with numbers (Section 14), it will fail to interpret such information in an appropriate manner.
- In principle, all such encoded information may be translated into a simpler kind of representation in which, for example, the brightness in any given portion of an image would be represented by the density

²See ‘Google Street View’ in ‘Google Maps’, 2018-05-11.

of ‘0’s or ‘1’s, much as in old-style black and white newspaper photograph [2, Section 2.2.3]. But it is not clear at present how the SP System would work in processing information of that kind.

In general, the SP System works best with SP-symbols representing structures or entities that are significant in human perception. Thus:

‘For the SP System to be effective in the processing of speech or visual images, it seems likely that some kind of preliminary processing will be required to identify low-level perceptual features, such as, in the case of speech, phonemes, formant ratios or formant transitions, or, in the case of visual images, edges, angles, colours, luminances or textures.’ [3, Section 3.3].

In broad terms, there are two ways in which the developing SP Machine may bridge the gap which is likely to exist between information that is drawn relatively directly from the environment and the kinds of structures just mentioned:

- As a stop-gap measure, preliminary processing of sensory data may be done by conventional methods to yield the kinds of low-level perceptual entities with which the SP System can operate.
- Given the power and generality of the principles that lie at the heart of the SP Theory, it seems likely that, pending some further research, the SP System, without stop-gap measures, will be able to process all kinds of information in a meaningful way, including ‘raw’ sensory data and data that has been encoded using numbers.

11. THE PROCESSING OF NATURAL LANGUAGE

A major strength of the SP-multiple-alignment framework is that it lends itself well to the representation and processing of natural language, as described quite fully in [2, Chapter 5] and more briefly in [3, Section 8]. Here are some examples:

- As can be seen in Fig. A3 in Appendix A.4, and in many other examples in [2, 3], the building of SP-multiple-alignments can achieve the effect of parsing a sentence into its parts and sub-parts.
- The SP-multiple-alignment concept exhibits versatility in the representation and processing of several kinds of non-syntactic knowledge (Appendix A.7). Any of these may represent the semantics or meanings of a sentence.
- As noted in Appendix A.7, the fact that one relatively simple framework may serve in the representation and processing of diverse kinds of knowledge means that there is potential for their seamless integration in any combination. More specifically in the present context,

there is clear potential for the seamless integration of the syntax and semantics of natural language.

- Preliminary examples described in [2, Section 5.7] show how this may work in both the semantic interpretation of a simple sentence [2, Figure 5.18] and in the production of a sentence from its meanings [2, Figure 5.19].
- The potential of the SP System for the seamless integration of diverse kinds of knowledge and diverse aspects of intelligence (Appendix A.7) means that there is clear potential for the integration of natural language processing with other aspects of intelligence such as pattern recognition, reasoning, planning, and problem solving.

In order to demonstrate parsing via SP-multiple-alignment, the semantic interpretation of a sentence, or the production of a sentence from meanings, it has been necessary to construct appropriate grammars by assembling collections of SP-patterns manually. This has been acceptable in the process of developing the SP-multiple-alignment framework and demonstrating what it can do, but it is likely to be far too slow and prone to error in the further development of natural language processing in the SP System.

To make progress with the processing of natural language in the SP System, it will be necessary to develop this area in conjunction with the development of unsupervised learning in the SP System, as described in the next section.

12. UNSUPERVISED LEARNING

In the light of observations in the previous section, the discussion of unsupervised learning in this section concentrates mainly on unsupervised learning of the syntax and semantics of natural language. Success with those aspects of natural language is likely to generalize to other kinds of knowledge and, in any case, ‘semantics’ is a broad concept that embraces most kinds of knowledge.

The emphasis on *unsupervised* learning in the SP Machine is because of the working hypothesis in this research that most human learning is achieved without the benefit of any kind of ‘teacher’ or anything equivalent (*cf.* [23]), and because of a further working hypothesis that other kinds of learning, such as supervised learning, reinforcement learning, learning by being told, and learning by imitation, may be seen as special cases of unsupervised learning [7, Sections V-A.1 and V-A.2].

The best place to start in developing unsupervised learning in the SP Computer Model is with two weaknesses in the SP71 model, outlined here:

‘A limitation of the SP Computer Model as it is now is that it cannot learn intermediate levels of abstraction in grammars (e.g. phrases and clauses), and it cannot learn the kinds of

discontinuous dependencies in natural language syntax that are described in [[3, Sections 8.1 to 8.3] and [2, Section 5.4]]. I believe these problems are soluble and that solving them will greatly enhance the capabilities of the system for the unsupervised learning of structure in data' [3, Section 3.3].

Pending further analysis, the problems with SP71 in discovering structures such as phrases and clauses in the syntax of natural language, and in discovering discontinuous dependencies in that syntax, may both be solved in a manner like what follows.

Since the building of SP-multiple-alignments is an integral part of unsupervised learning in the SP System (Appendix A.5.3), and since every stage in the process of building an SP-multiple-alignment yields an encoding of one or more New SP-patterns in terms of the Old SP-patterns in the SP-multiple-alignment, *better results may be obtained by applying the learning processes to those encodings as well as to the original SP-patterns.*

Probably, the best way to build up an understanding of what needs to be done is to work with unsegmented samples of relatively simple English-like languages generated by artificial grammars. If or when good insights and good results can be obtained with that kind of material, the system may be tested and refined with more challenging material, as described in the next three subsections.

12.1. Learning the syntax of natural language

Following the steps outlined above, it seems possible now to bring the SP System to a stage of development where it can derive plausible grammars for the syntax of natural language via the application of unsupervised learning to unsegmented textual samples of natural language. This would be a major achievement since it appears that no other system can perform at that level.

Incidentally, for reasons given in [3, Section 6.2], there are likely to be subtle but important differences between grammars derived from natural language texts without meanings and the kinds of grammars that people learn as children, where syntax and semantics are developed in tandem.

It seems likely that success with this task would smooth the path for success with the tasks described next.

12.2. The unsupervised learning of non-syntactic kinds of knowledge

A key idea in the SP programme of research is that one set of principles applies across the representation and processing of diverse kinds of knowledge.

This idea, which was originally a working hypothesis motivated by Ockham's Razor, is now backed by substance in the form of the concept of SP-multiple-alignment and its

versatility in support of diverse aspects of intelligence and its versatility in the representation of diverse forms of knowledge (Appendix A.7). In general, the SP System has potential to be *a universal framework for the representation and processing of diverse kinds of knowledge* (UFK) [6, Section III].

Hence, it is anticipated that if good solutions can be found for the unsupervised learning of syntax, it is likely that, with or without some further development, they would generalize to the learning of non-syntactic, semantic structures.

With the learning of visual structures, it would be necessary for there to be a robust solution to the previously mentioned problem of finding good full and partial matches between 2D SP-patterns (Section 9.1).

12.3. Learning syntactic/semantic structures

A prominent feature of the environment in which most young children learn their first language or languages is that, very often, they can hear the speech of adults or older children at the same time as they can see or hear what people are talking about: 'Lunch is on the table', 'Here's our bus coming', and so on.

It appears that this kind of environment is needed to enable young children to work out the meanings of words and grammatical forms, and something similar would be needed for experiments with the SP System. But there appears to be a problem:

'The logician W. V. O. Quine asks us to imagine a linguist studying a newly discovered tribe. A rabbit scurries by, and a native shouts, 'Gavagai!' What does gavagai mean? Logically speaking, it needn't be 'rabbit.' It could refer to that particular rabbit (Flopsy, for example). It could mean any furry thing, any mammal, or any member of that species of rabbit (say, *Oryctolagus cuniculus*), or any member of that variety of that species (say, chinchilla rabbit). It could mean scurrying rabbit, scurrying thing, rabbit plus the ground it scurries upon, or scurrying in general.... Somehow a baby must intuit the correct meaning of a word and avoid the mind-boggling number of logically impeccable alternatives. It is an example of a more general problem that Quine calls 'the scandal of induction,' which applies to scientists and children alike: how can they be so successful at observing a finite set of events and making some correct generalization about all future events of that sort, rejecting an infinite number of false generalizations that are also consistent with the original observations?' [24, pp. 147–148].

Without wishing to trivialize this problem, it appears that the SP System has potential to provide a good solution:

- By searching for recurrent patterns and attaching more weight to patterns that occur frequently and less weight to patterns that are rare, the SP System naturally homes in on the stronger correlations—such as the correlation between the word 'lunch' and what people eat at lunchtime—and discards weaker

correlations—such as the correlation between the word ‘lunch’ and whether or not it is raining outside.

- The problem of how to generalize from what one has learned, without either under-generalization or over-generalization, is discussed in [3, Section 5.3]³ and also in [4, Section V-H].
- In brief, learning with the SP System from a body of data, **I**, means compressing **I** as much as possible and then splitting the resulting body of data into two parts: a *grammar* which contains a single copy of each recurrent SP-pattern in **I**; and an *encoding* of **I** which contains non-recurrent parts of **I** and brief codes or references to recurrent SP-patterns in **I**.⁴

Evidence to date shows that with learning of this kind, the grammar generalizes in a way that native speakers judge to be ‘correct’ and avoids what intuitively such people would judge to be under-generalization or over-generalization.⁵

Assuming that progress can be made along the lines outlined here, another avenue that may be explored later is how children may learn the meanings of things by making inferences from their existing knowledge. For example, if a child that does not know the word ‘plantain’ hears someone say something like: ‘I had plantain for lunch today. It was delicious—more tasty than potato but not overpowering’, he or she may infer quite reasonably that plantain is something that one can eat and that it is probably something starchy, like potato.⁶

In this area of research—how the syntax and semantics of natural language may be learned together—it will be necessary to provide input to the SP System that is comparable with that described above.

Also, it will probably be necessary for the SP System to have been generalized to work with 2D SP-patterns (Section 9.1), so that syntactic knowledge may be shown alongside semantic knowledge, with associations between them (see also discussions in [7, Sections V-G]).

12.4. Learning and use of causal relations

Since concepts of causation and how they may be learned have been the subject of much research and discussion by, for example, Judea Pearl [25, 26], it will not be possible here to do more than make a few tentative remarks and suggestions.

In view of the versatility of the SP System in aspects of intelligence and in the representation of knowledge (Appendix

A.7), and in view of the central role of information compression in the workings of the SP System (Appendix A.1 and Appendix A.3), it seems likely that information compression will also have an important explanatory role in the representation and learning of causal relations.

Although superficially there may seem to be little in common between information compression and causal relations, the connection may make more sense in the light of the intimate connection that is known to exist between information compression and concepts of correlation, inference, and probability (Appendix A.2).

But it is well known that finding a correlation between two things, such as smoking and cancer, is not the same as demonstrating a causal relation. A useful example in this connection is that two radios, R1 and R2, that are tuned to the same transmitter, T, will produce sounds that are perfectly correlated, but it is clear that neither radio has a causal connection with the other. Of course, it is T which is the real driver of both R1 and R2.

Examples like these seem, superficially, to invalidate any scheme that attempts to explain the learning and representation of causal relations in terms of correlation and closely related concepts of information compression.

But: a full understanding of the relationships amongst T, R1, and R2 may be regarded as a ‘deeper’ understanding than one that merely sees the correlation between R1 and R2; and that intuitive concept of ‘depth’ of understanding may turn out to be expressible in terms of levels of information compression; then, it may prove feasible in the SP System to learn causal relations via information compression, in much the same way that other kinds of knowledge may be learned via compression of information.

12.5. Learning language from speech

Although, to a large extent, children learn their first language or languages by hearing people talk, learning a language via speech has in this paper been left to last because it is more challenging than, for example, learning grammars from text.

The main reason for the difficulty is uncertainties about the discovery of low-level features in speech, as discussed in Section 10. Solutions in that area will facilitate the learning of language from speech.

12.6. Learning from data in two and three dimensions

As noted in Section 9.1 a significant challenge will be learning from SP-patterns in two dimensions, and as noted in Section 9.3, an important application of the SP System will be learning, from videos or the like, 2D and 3D structures that change across the time dimension, and with sounds including speech associated with vision.

³Under-generalization was overlooked in this account but the avoidance of under-generalizations is achieved in exactly the same way as the avoidance of over-generalizations.

⁴Notice that recurrent SP-patterns in **I** may be abstract patterns as well as concrete patterns.

⁵See also the discussion of generalization in [13].

This last kind of learning approximates what we all do in our learning from birth onwards. It may be seen as a ‘holy grail’ for learning—something which will be necessary but not sufficient to realize the full potential of the SP System.

13. COMPUTER VISION

The generalization of the SP Machine to work with 2D SP-patterns (Section 9.1) is a likely prerequisite for the development of the SP System for computer vision and as a model of human vision, as described in [21].

An important part of this development will be the unsupervised learning or discovery of objects and classes, as discussed in [21, Section 5] and Section 12.2, below. This will include work on the discovery of low-level features in images, as discussed in Section 10.

14. PROCESSING NUMBERS

In view of evidence that mathematics may be seen as a set of techniques for compression of information, and their application [27], and bearing in mind that the SP System works entirely via the compression of information, and in view of the potential of the SP System as a vehicle for ‘computing’ ([2, Chapter 4], [5, Section 6.6], [28]), and as a vehicle for mathematics [2, Chapter 10], there is clear potential in the SP Machine for the performance of arithmetic and related computations. But the SP Computer Model does not yet handle numbers in a satisfactory way.

Does this matter? In terms of short-term practicalities, the answer is ‘no’—because it would be easy enough to provide some kind of maths co-processor within the SP Machine that would handle numbers and arithmetic operations very well. But in terms of the long-term objectives of the SP programme of research—creating a system with a favourable combination of conceptual *simplicity* with descriptive and explanatory *power*—the use of a maths co-processor would be an *ad hoc* development, much like a software ‘hack’ without good design.

We believe that, as far as possible, the long-term objectives of the SP research should be maintained (Section 3.1). This strategy has yielded substantial dividends already and we believe there is much more to come.

But pending more mature capabilities for arithmetic in the SP Machine, there may be a case for using an existing maths coprocessor or similar facility as an adjunct to the SP Machine, much as is done in many database management systems today.

15. SP-NEURAL

At present, the SP-Neural part of the SP Theory (Appendix A.6) exists only as a conceptual model, described in some detail in [29].

To flesh out this conceptual model, it is really necessary to create a computational version of it. As with the SP71 computer model, this will help to guard against vagueness in the theory, it will serve as a means of testing ideas to see whether or not they work as anticipated, and it will be a means of demonstrating what the model can do.

The best place to start would be to create a computer model of the building of an SP-multiple-alignment in SP-Neural, as discussed in [29, Section 4] and illustrated schematically in Fig. 2.

The development of unsupervised learning in the computer model of SP-Neural is probably best left until later, if or when there is a satisfactory SP-Neural model of the building of the neural equivalent of SP-multiple alignments. This is because the building of SP-multiple alignments is an important part of unsupervised learning in the SP System. There are some relevant remarks about unsupervised learning in Section 15.3.

In Fig. 2, each alphanumeric symbol surrounded by a solid-line envelope represents a *neural symbol* (Appendix A.6), and each group of neural symbols surrounded by a broken-line envelope represents a *pattern assembly* (Appendix A.6, [29, Section 4.2]).

In the figure, nerve-fibre connections between neural symbols are represented with lines which may be solid or broken. The solid lines represent neural connections in part of the SP-multiple-alignment shown in [29, Figure 2], while the broken lines represent other neural connections that are not part of that SP-multiple-alignment.

15.1. Inhibition

This section considers the potential role of inhibitory processes in the new computer model, starting with some background.

15.1.1. Background

In [29, Section 9], there is a brief review of what appears to be a substantial role for inhibitory signals between neurons in the workings of brains and nervous systems. A tentative general rule from this research is that *When, in neural processing, two or more signals [that interact] are the same, they tend to inhibit each other, and when they are different, they don’t* [29, p. 19], with the implication that ‘The overall effect should be to detect redundancy in information and to reduce it, whilst retaining non-redundant information, in accordance with the central principle in the SP Theory—that much of computing and cognition may, to a large extent, be understood as information compression.’ (*ibid.*).

A related idea is that the rate of firing of neurons correlates only indirectly with the strength of stimulation, and that when stimulation is steady, neurons typically fire at a constant rate which is the same regardless of the strength of the stimulation. This can be seen in Fig. 3 which shows variations in the

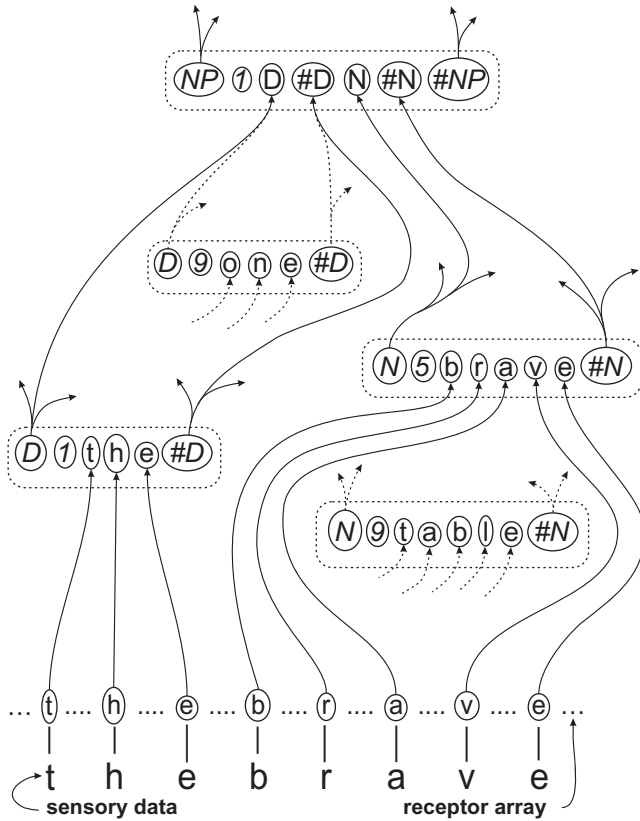


FIGURE 2. A schematic representation of part of the process of building an SP-multiple-alignment in SP-Neural, as discussed in [29, Section 4]. Each broken-line rectangle with rounded corners represents a *pattern assembly*—corresponding to a ‘pattern’ in the main SP Theory; each character or group of characters enclosed in a solid-line ellipse represents a *neural symbol* corresponding to a ‘SP-symbol’ in the main 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 Fig. 3 in [29].

rate of firing of a sensory cell (*ommatidium*) in the eye of the horseshoe crab (*Limulus*) in response to variations in the intensity of the light that is falling on it.

On the left of the figure, where the light level is low, the ommatidium fires at a steady rate of about 20 impulses per second. When a light is switched on, there is sharp upswing in the rate of firing, but this soon settles back to the same 20 impulses per second rate as before. When the light is switched off, there is a sharp downswing in the rate of firing followed by a rapid return to the previous steady rate.

This pattern of responses can be explained via inhibitory connections which dampen down responses when stimulation

is steady [30, 31].⁶ Similar effects occur in the spatial dimension and it appears that similar explanations apply.

15.1.2. Developments

In the building of an SP-multiple-alignment, modelling such things as the parsing of a sentence or the recognition of a pattern, things are a bit different from what has been described in the previous subsection. Instead of trying to detect and remove redundancy between neighbouring elements in a sequential or spatial pattern, the aim is to detect redundancies between two independent patterns, such as a New SP-pattern (as, for example, ‘I N F O R M A T I O N’) and an Old SP-pattern (such as ‘N 6 I N F O R M A T I O N #N’), or between two partial SP-multiple-alignments at later stages of processing.

Bearing in mind the putative idea that the role of inhibition is to detect and remove redundancies in information, a possible scheme is shown in Fig. 4, with a key to the meanings of SP-symbols in the caption to the figure.

This is how the scheme may work:

- In accordance with what was said about the results of neural processing shown in Figure 3, it is envisaged that, without incoming signals, the four neural symbols shown at the top of the figure will fire at a steady intermediate rate.
- If a signal comes in from the neural symbol marked ‘a’ at the bottom of the figure, this will have an inhibitory effect on the neural symbol marked ‘a’ at the top of the figure. Likewise for the other two neural pairings, ‘b’ with ‘b’, and ‘c’ with ‘c’.
- If all three of ‘a’, ‘b’, and ‘c’, in the pattern assembly have been inhibited in this way, there will be reduced inhibitory signals transmitted to the ‘identifier’ (ID) neural symbol marked ‘X’ in the figure.
- Since the ID neural symbol is receiving reduced inhibitory signals, its rate of firing will increase. This may be interpreted as recognition of the pattern ‘a b c’. That increase in the rate of firing may be transmitted to higher level pattern assemblies that contain ‘X’ as a ‘reference’ to the pattern ‘a b c’. This may lead to similar processes of recognition at higher levels.

A possible weakness in the account just given is that, in line with what was said about Figure 3, it may be supposed that the rate of firing of the ‘X’ neural symbol would quickly settle back to the average rate of firing, but there is nothing to ensure that would happen. Issues like this may be clarified with a working model.

What about the ordering of neural symbols? With the scheme as just described, the pattern assembly ‘X a b c’

⁶Although the two sources that have been referenced have authority in this area, they do not explain the principles very well. A clearer account may be found in [32, pp. 65–75].

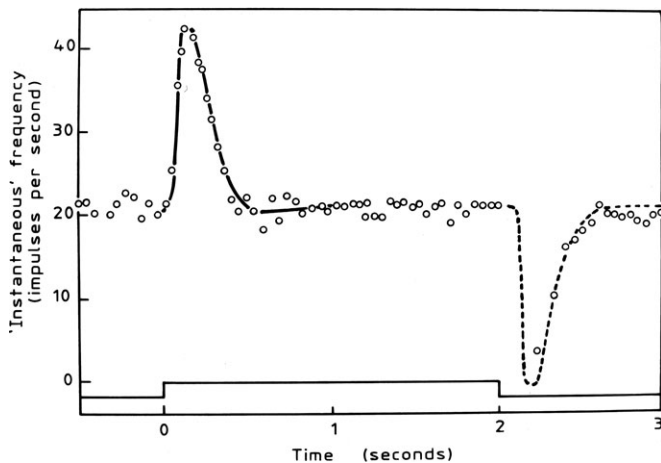


FIGURE 3. Variation in the rate of firing of a single ommatidium of the eye of a horseshoe crab in response to changing levels of illumination. Reproduced from [30, Figure 16], with permission from the Optical Society of America.

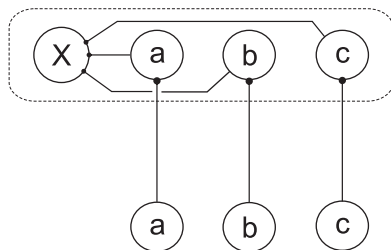


FIGURE 4. A schematic representation of one possible neural scheme for detecting redundancies between an incoming pattern ('a b c'), shown at the bottom of the figure and a stored pattern that matches it, shown above. Key: a letter inside an unbroken envelope represents a neural symbol; a group of neural symbols surrounded by a broken-line envelope represents a pattern assembly; lines between neural symbols ending in a blob (a filled circle) represent nerve fibres carrying signals towards the blob, with an inhibitory impact on the neural symbol that is receiving the signals.

would respond as strongly to an input pattern 'b c a' as it would to the input pattern 'a b c'. But it looks as if the concept of a *typographic pattern*, provides an answer. As was quoted in [29, Section 5.4]:

'Receptors in the retina and body surface are organized as two-dimensional sheets, and those of the cochlea form a one-dimensional line along the basilar membrane. Receptors in these organs communicate with ganglion cells and those ganglion cells with central neurons in a strictly ordered fashion, such that relationships with neighbours are maintained throughout. **This type of pattern, in which neurons positioned side by side in one region communicate with neurons positioned side-by-side in the next region, is called a typographic pattern.**' [33, p. 504], *italic emphasis in the original, boldface added.*

What this concept of a 'typographic pattern' would mean for the workings of the scheme shown in Figure 4 is that the neural symbol 'X' is a detector for the pattern 'a b c' and not for 'b c a', 'c b a', and so on, because it only ever receives signals in the order 'a b c'.

15.2. Implementation of the SP-Neural Computer Model

The SP-Neural Computer Model may be developed with these main features:

- *The C++ computer language.* To be compatible with the existing SP Computer Model (SP71), the model is probably best implemented using the C++ computer language.
 - *Neural symbols as a class.* Neural symbols would be implemented as a C++ class. For each instance of the class, this would provide the following features:
 - *Inhibitory connections between neurons.* The paths for inhibitory messages to one or more neural symbols would be nerve-fibre connections like the one between neural symbol 'a' near the bottom of Figure 4 and neural symbol 'a' near the top, and likewise for the two other pairs of neural symbols in the figure.
 - *Indexing of symbols that match each other.* If the nerve-fibre connections just mentioned are only ever between a pair of symbols that match each other, such as, for example, between 'N' as the ID neural symbol in a pattern assembly like 'N d o g' (the word 'dog' belongs in the class 'noun'), and 'N' as part of the contents of a pattern assembly like 'S N V' (a sentence in its simplest form is a noun followed by a verb), then the list of connections in each neural symbol is, in effect, an index of matching symbols, much like the index proposed for the abstract part of the SP Theory, described in Section 5.
- This line of thinking suggests that many neural connections in the brain, perhaps all of them, are, in effect, direct connections between neural symbols with a role that is very much like that of an index in any ordinary database management system, or the very large indices that are maintained in search engines for the Internet.
- *Inhibition of ID neural symbols.* Within each pattern assembly, each neural symbol would have the means of sending an inhibitory message to ID neural symbol at the far left of the pattern assembly, as shown in Figure 4. Notice that any one neuron can only ever be in one pattern assembly, or none. Hence, any one neuron only ever needs to be able to send inhibitory messages to an ID neural symbol in the pattern assembly of which it is a part.

- *Subclasses of the neural symbol class.* There may be a case for providing two subclasses of the neural symbol class: one for *C neural symbols* (each of which may be part of the contents of a pattern assembly) and one for *ID neural symbols* (each of which may serve as an identifier for a pattern assembly). These features of SP-Neural are described in [2, Section 11.3.2].⁷
- *Pattern assemblies as a class.* Pattern assemblies would also be implemented as a C++ class. For each instance of the class, it would provide the means of storing zero or more ID-neural-symbols and zero or more C-neural-symbols, perhaps using linked lists. It may also store other information such as the number of times the pattern assembly has been recognized in a given set of New SP-patterns.

15.3. Unsupervised learning in SP-Neural

How unsupervised learning may be achieved in SP-Neural is considered mainly in [29, Section 10] with some related observations in Section 11 of that paper. Here, there are no new insights to be added to that account except to underscore the point that:

‘... in any or all of short-term memory, working memory, and long-term memory, SP-Neural may achieve the necessary speed in the creation of new structures, combined with versatility in the representation and processing of diverse kinds of knowledge, by the switching on and off of synapses in pre-established neural structures and their interconnections ...’ and that this would be ‘somewhat like the way in which an ‘uncommitted logic array’ (ULA) may, via small modifications, be made to function like any one of a wide variety of ‘application-specific integrated circuits’ (ASICs), or how a ‘field-programmable gate array’ (FPGA) may be programmed to function like any one of a wide variety of integrated circuits’ [29, Section 11].

16. NEW HARDWARE

As was mentioned in the Introduction, there may, at some stage, be a case for the development of new hardware, dedicated to structures and processes in the SP System, and with potential for gains in efficiency and performance ([6], [7, Section IX]).

Since searching for matching SP-patterns is an important part of how the SP System works, gains in efficiency may be

achieved by concentrating search where good results are most likely to be found: ‘If we want to find some strawberry jam, our search is more likely to be successful in a supermarket than it would be in a hardware shop or a car-sales showroom.’ [6, Section IX-A.2], and the statistical knowledge in the system flows directly from the central role of information compression in the workings of the SP System, and from the intimate relationship that is known to exist between information compression and concepts of correlation, inference and probability (Appendix A.2).

Gains in efficiency from the use of indices (Section 5) may be at least partly via the encoding of statistical aspects of the data that are being processed.

New hardware may be developed for either or both of the two versions of the SP System: the abstract version embodied in the SP71 computer model, and SP-Neural as outlined in Appendix A.6 and discussed in Section 15.

17. DEVELOPMENT OF APPLICATIONS OF THE SP MACHINE

Sources of information about potential applications of the SP Machine are detailed in Appendix A.8. Each one of these potential applications may be the subject of one or more research projects in its own right.

18. APPLICATIONS THAT MAY BE REALIZED QUITE SOON

In case practical applications of the SP Machine seem too distant to be supported, some of them may be realized quite soon. Here, in brief, are some examples:

- *The SP Machine as an intelligent database system.* With the application of parallel processing and indexing (Section 4) and the creation of a ‘friendly’ user interface (Section 7), the SP Machine should begin to be useful as an intelligent database system, with several advantages over ordinary database systems, as described in [14].
- *Information compression.* Since information compression is central in how the SP System works, the SP Machine is likely to prove useful in situations where compression of information is the main requirement. It may, for example, be a useful means of reducing the size of ‘big data’, with corresponding benefits in storage and transmission of those data [6, Section VII].
- *Several kinds of reasoning.* Most of the several kinds of reasoning described in [2, Chapter 7] will be available immediately in the highly parallel version of the SP Machine. Because they all flow from one simple framework—SP-multiple-alignment—they may be applied in any combination (meaning all the several

⁷The term *C-neuron* (meaning ‘C neural symbol’), introduced in [2, Section 11.3.2], and the term *ID-neuron* (meaning ‘ID neural symbol’), introduced in the same place, should probably be dropped. This is because neural symbols are not necessarily single neurons—they may be small clusters of neurons.

kinds of reasoning or any subset of them), and with seamless integration among them.

- *Model-based coding for the efficient transmission of information.* The concept of model-based coding was described by John Pierce in 1961 like this:

‘Imagine that we had at the receiver a sort of rubbery model of a human face. Or we might have a description of such a model stored in the memory of a huge electronic computer. First, the transmitter would have to look at the face to be transmitted and ‘make up’ the model at the receiver in shape and tint. The transmitter would also have to note the sources of light and reproduce these in intensity and direction at the receiver. Then, as the person before the transmitter talked, the transmitter would have to follow the movements of his eyes, lips and jaws, and other muscular movements and transmit these so that the model at the receiver could do likewise.’ [34, pp. 139–140].

At the time this was written, it would have been impossibly difficult to make things work as described. But it seems likely that, with the SP System, it will become feasible fairly soon in a simple but effective form, and it will improve progressively with improvements in unsupervised learning in the SP System.

19. CONCLUSION

This paper describes a roadmap for the development of a mature version of the SP Machine, starting with the SP71 computer model. After an overview of the strategy adopted for this programme of research, there is a description of the main tasks that are to be tackled, probably in the following order:

- (1) Starting with the SP71 computer model, create a highly parallel version of the SP Machine as a software virtual machine hosted on a high-performance computer. An interesting alternative would be to create the SP Machine as a software virtual machine driven by highly parallel search processes in any of the leading search engines.
- (2) Create a user interface for the SP Machine that is easy to use.
- (3) Closely related to the previous task is the development of the SP System’s existing strengths in the visualization of the structure of stored knowledge and in the provision of an audit trail for all processing.
- (4) As the basis for such things as computer vision, the SP Machine will need to be generalized to work with SP-patterns in two dimensions as well as the 1D SP-patterns of the SP71 model.

- (5) The generalization just mentioned would include the development of processes for the discovery of good full and partial matches between 2D SP-patterns (and between a 2D SP-pattern and a 1D SP-pattern) and for the display of SP-multiple-alignments comprising two or more 2D SP-patterns (with or without 1D SP-patterns).
- (6) An investigation is needed into whether or how the SP Machine, with or without some modification, may discover low-level features in speech and visual images, of the kind that appear to be significant in human perception.
- (7) Existing strengths of the SP System in the processing of natural language may be developed towards the goal of creating an SP Machine that can understand natural language and produce natural language from meanings.

This is likely to be a major project or programme of research, with several stages, in conjunction with the development of unsupervised learning (next bullet point).

Likely stages include: experiments with the unsupervised learning of artificial languages; unsupervised learning of plausible grammars from unsegmented samples of natural language text; unsupervised learning of plausible structures for various kinds of non-syntactic knowledge; unsupervised learning of syntactic/semantic structures; and unsupervised learning from speech.

- (8) Residual problems with the existing processes for unsupervised learning need to be solved. These weaknesses are a failure of the system to discover grammatical structures with intermediate levels of abstraction, and failure to discover discontinuous dependencies in such structures.
A solution to these problems will clear the path for the development of unsupervised learning in conjunction with the processing of natural language, as described above.
- (9) Existing strengths of the SP System in pattern recognition may be developed for computer vision, guided by the insights in [21].
- (10) On the strength of evidence to date, it seems likely that the representation of numbers and the performance of arithmetic processes may be accommodated within the SP framework. Pending success in those areas, stop-gap solutions may be employed where needed.
- (11) Somewhat independent of other developments described in this paper, would be the development of a computer model of SP-Neural, drawing on the information and insights described in [29] and perhaps also in [2, Chapter 11].

An important part of this development would be to see whether or how inhibitory processes have the important role that seems likely from evidence that is available now, and, if not, to provide some reasons why.

- (12) At some stage when the SP Machine is relatively mature, it is envisaged that new hardware would be developed, mainly to exploit opportunities to increase the efficiency of computations, most notably by taking advantage of statistical information that the SP System gathers as a by-product of how it works.

There is potential for the SP Machine to be applied on relatively short timescales in such areas of application as information storage and retrieval, information compression, several kinds of reasoning and model-based coding for the efficient transmission of information.

ACKNOWLEDGEMENTS

We are grateful to anonymous reviewers for many helpful suggestions and corrections.

REFERENCES

- [1] Wolff, J.G. (1990) Simplicity and power—some unifying ideas in computing. *Comput. J.*, **33**, 518–534.
- [2] Wolff, J.G. (2006) *Unifying Computing and Cognition: The SP Theory and Its Applications*. CognitionResearch.org, Menai Bridge. Distributors, including Amazon.com, are detailed on bit.ly/WmB1rs.
- [3] Wolff, J.G. (2013) The SP theory of intelligence: an overview. *Information*, **4**, 283–341.
- [4] Wolff, J.G. (2016) The SP theory of intelligence: its distinctive features and advantages. *IEEE Access*, **4**, 216–246.
- [5] Wolff, J.G. (2014) The SP theory of intelligence: benefits and applications. *Information*, **5**, 1–27.
- [6] Wolff, J.G. (2014) Big data and the SP theory of intelligence. *IEEE Access*, **2**, 301–315.
- [7] Wolff, J.G. (2014) Autonomous robots and the SP theory of intelligence. *IEEE Access*, **2**, 1629–1651.
- [8] McCorduck, P. (2004) *Machines Who Think: A Personal Inquiry Into the History and Prospects of Artificial Intelligence* (2nd edn). A. K. Peters Ltd, Natick, MA.
- [9] Newell, A. (1973) You can't play 20 questions with nature and win: Projective comments on the papers in this symposium. In Chase, W.G. (ed.) *Visual Information Processing*, pp.283–308. Academic Press, New York.
- [10] Newell, A. (ed.) (1990) *Unified Theories of Cognition*. Harvard University Press, Cambridge, Mass.
- [11] Everitt, T., Goertzel, B. and Potapov, A. (eds.) (2017) *Artificial General Intelligence Lecture Notes in Artificial Intelligence*. Springer, Heidelberg.
- [12] Langley, P. (2012) The cognitive system paradigm. *Adv. Cogn. Syst.*, **1**, 3–13.
- [13] Wolff, J.G. (1988) Learning syntax and meanings through optimization and distributional analysis. In Levy, Y., Schlesinger, I.M. and Braine, M.D.S. (eds.) *Categories and Processes in Language Acquisition*, pp.179–215. Lawrence Erlbaum, Hillsdale, NJ. bit.ly/ZIGjyc.
- [14] Wolff, J.G. (2007) Towards an intelligent database system founded on the SP theory of computing and cognition. *Data Knowl. Eng.*, **60**, 596–624. bit.ly/1CUldR6.
- [15] Schmidhuber, J. (2015) Deep learning in neural networks: an overview. *Neural Netw.*, **61**, 85–117.
- [16] Lampert, L. (1979) How to make a multiprocessor computer that correctly executes multiprocess programs. *IEEE Trans. Comput.*, **C-28**, 690–691.
- [17] Abdallah, A.E., Jones, C.B. and Sanders, J.W. (2005) *Communicating Sequential Processes: The First 25 Years Lecture Notes in Computer Science*, Vol. **3525**. Springer, Heidelberg.
- [18] Dean, J. and Ghemawat, S. (2008) Mapreduce: simplified data processing on large clusters. *Commun. ACM*, **51**, 107–113.
- [19] Lam, M. (1988) Software pipelining: an effective scheduling technique for vliw machines. *Proc. ACM SIGPLAN 1988 Conf. Programming Language Design and Implementation (PLDI '88)*, pp. 318–328. ACM.
- [20] Thesen, A. (1978) *Computer Methods in Operations Research*. Academic Press, New York.
- [21] Wolff, J.G. (2014) Application of the SP theory of intelligence to the understanding of natural vision and the development of computer vision. *SpringerPlus*, **3**, 552–570.
- [22] Wolff, J.G. (2006) Medical diagnosis as pattern recognition in a framework of information compression by multiple alignment, unification and search. *Decis. Support Syst.*, **42**, 608–625.
- [23] Gold, M. (1967) Language identification in the limit. *Inf. Control*, **10**, 447–474.
- [24] Pinker, S. (1995) *The Language Instinct* (Kindle edn). Penguin Books, London.
- [25] Pearl, J. (2000) *Causality: Models, Reasoning and Inference*. Cambridge University Press, Cambridge.
- [26] Pearl, J. and Mackenzie, D. (2018) *The Book of Why: The New Science of Cause and Effect* (Kindle edn). Allen Lane, London.
- [27] Wolff, J.G. (2017) Mathematics as information compression via the matching and unification of patterns. Technical Report. CognitionResearch.org. Submitted for publication. bit.ly/2LWbjtK
- [28] Wolff, J.G. (2017) Software Engineering and the SP Theory of Intelligence. Technical Report. CognitionResearch.org. Submitted for publication, arXiv:1708.06665 [cs.SE], bit.ly/2w99Wzq.
- [29] Wolff, J.G. (2016) Information compression, multiple alignment, and the representation and processing of knowledge in the brain. *Front. Psychol.*, **7**, 1584.
- [30] Ratliff, F., Hartline, H.K. and Miller, W.H. (1963) Spatial and temporal aspects of retinal inhibitory interaction. *J. Optical Soc. Am.*, **53**, 110–120.
- [31] von Békésy, G. (1967) *Sensory Inhibition*. Princeton University Press, Princeton, NJ.

- [32] Lindsay, P.H. and Norman, D.A. (eds.) (1975) *An Introduction to Psychology*. Academic Press, New York.
- [33] Squire, L.R., Berg, D., Bloom, F.E., du Lac, S., Ghosh, A. and Spitzer, N.C. (eds.) (2013) *Fundamental Neuroscience* (4th edn). Elsevier, Amsterdam.
- [34] Pierce, J.R. (1961) *Symbols, Signals and Noise*. Harper & Brothers, New York.
- [35] Wolff, J.G. (2017) Information Compression via the Matching and Unification of Patterns as a Unifying Principle in Human Learning, Perception, and Cognition. Technical Report. CognitionResearch.org. Submitted for publication. bit.ly/2ruLnrV.
- [36] Shannon, C.E. and Weaver, W. (1949) *The Mathematical Theory of Communication*. University of Illinois Press, Urbana.
- [37] Solomonoff, R.J. (1964) A formal theory of inductive inference. Parts I and II. *Inf. Control*, **7**, 1–22. and 224–254.
- [38] Li, M. and Vitányi, P. (2014) *An Introduction to Kolmogorov Complexity and Its Applications* (3rd edn). Springer, New York.
- [39] Wagner, R.A. and Fischer, M.J. (1974) The string-to-string correction problem. *J. ACM*, **21**, 168–173.
- [40] Sankoff, D. and Kruskal, J.B. (1983) *Time Warps, String Edits, and Macromolecules: the Theory and Practice of Sequence Comparisons*. Addison-Wesley, Reading, MA.
- [41] Hebb, D.O. (1949) *The Organization of Behaviour*. John Wiley & Sons, New York.
- [42] Kirkpatrick, J. (2017) Overcoming catastrophic forgetting in neural networks. *Proc. Natl Acad. Sci. USA*, **114**, 3521–3526.
- [43] Wolff, J.G. (2018) Solutions to Problems with Deep Learning. Technical Report. CognitionResearch.org. arXiv:1801.05457 [cs.LG], bit.ly/2AJzu4j.
- [44] Wolff, J.G. (2018) Introduction to the SP Theory of Intelligence. Technical Report. CognitionResearch.org. arXiv:1802.09924, bit.ly/2ELq0Jq.
- [45] Wolff, J.G. (2016) Commonsense Reasoning, Commonsense Knowledge, and the SP Theory of Intelligence. Technical Report. CognitionResearch.org. Submitted for publication, arXiv:1609.07772 [cs.AI], bit.ly/2eBoE9E.
- [46] Davis, E. and Marcus, G. (2015) Commonsense reasoning and commonsense knowledge in artificial intelligence. *Commun. ACM*, **58**, 92–103.

APPENDIX A. OUTLINE OF THE SP SYSTEM

To help ensure that this paper is free standing, the SP System is described in outline here with enough detail to make the rest of the paper intelligible.

The SP System is described most fully in [2] and more briefly in [3].

The SP Theory is conceived as a brain-like system as shown schematically in Figure A1. The system receives *New* information via its senses and stores some or all of it in compressed form as *Old* information.

A.1. Information Compression via the Matching and Unification of Patterns

Information compression is central in the SP System because of substantial evidence for the importance of information compression in the workings of brains and nervous systems described in [35].

Since people often ask what the meaning is of the name ‘SP’, it is short for *Simplicity* and *Power*. This is because information compression may be seen to be a process of maximizing the *simplicity* of a body of information, **I**, by extracting redundancy from **I**, while retaining as much as possible of its non-redundant descriptive *power*.

As the title of this subsection suggests, information compression in the SP System is not any kind of information compression, it is information compression that is achieved via a search for patterns that match each other and the merging or ‘unification’ of patterns that are the same. This orientation has been adopted because it seems to provide a better handle on possible mechanisms for the compression of information in natural or artificial systems than do the more mathematically-oriented approaches to information compression.

A.2. Information Compression and Concepts of Correlation, Inference and Probability

It has been known for some time that there is an intimate connection between information compression and concepts of correlation, inference and probability [36–38]. Hence, dedication of the SP System to information compression means that, in the SP System, it is relatively straightforward to make inferences and to calculate associated probabilities.

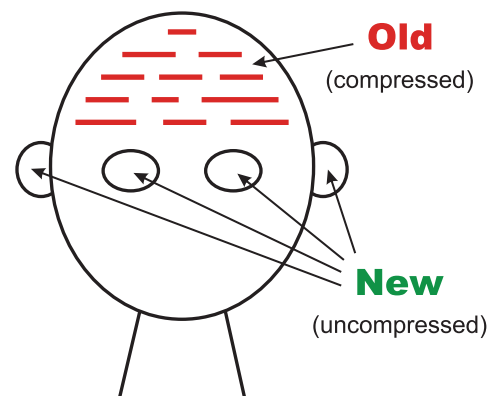


FIGURE A1. Schematic representation of the SP System from an ‘input’ perspective. Reproduced with permission from Fig. 1 in [3].

A.3. SP-multiple-alignments

In the SP System, information compression is achieved largely via the concept of *SP-multiple-alignment*, a concept that has been adapted from the concept of ‘multiple sequence alignment’ in bioinformatics.

As illustrated in Figure A2, a multiple sequence alignment in bioinformatics is an arrangement of two or more DNA sequences or amino-acid sequences so that, by judicious ‘stretching’ of sequences in a computer, symbols that match each other are brought into line. A ‘good’ multiple sequence alignment is one in which a relatively large number of matching symbols have been aligned.

Because there is normally an astronomically large number of alternative ways in which multiple sequence alignments may be arranged, heuristic methods are needed to build multiple sequence alignments in stages, discarding all but the best few partial multiple sequence alignments at the end of each stage. With such methods, it is not normally possible to guarantee that a theoretically ideal multiple sequence alignment has been found, but it is normally possible to find multiple sequence alignments that are ‘reasonably good’.

```

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 A2. A ‘good’ multiple sequence alignment among five DNA sequences. Reproduced with permission from Fig. 3.1 in [2].

The following subsections describe how the multiple sequence alignment concept has been adapted to become the concept of SP-multiple-alignment.

A.4. SP-patterns, SP-symbols, and SP-multiple-alignments in the SP System

In the SP System, *all* kinds of knowledge are represented with arrays of atomic *SP-symbols* in one or two dimensions, termed *SP-patterns*. Of course, a 1D SP-pattern is the same as a sequence in bioinformatics, but in the SP System the concept of an SP-pattern includes both 1D and 2D arrays of symbols. Although, at present, the SP Computer Model works only with 1D SP-patterns, it is envisaged that it will be developed to work with 2D SP-patterns as well (Section 4).

There may also be a case for introducing unordered SP-patterns (sets) in which the order of the symbols is not significant—but that effect may be achieved via the unordered nature of sets of New SP-patterns (*cf.*, [22, Section 3.4]).

In themselves, SP-patterns are not very expressive. But within the SP-multiple-alignment framework they become very versatile as a means of representing diverse kinds of knowledge [28, Appendix B.1].

The main change to the multiple sequence alignment concept in the SP System is that, in each SP-multiple-alignment, one or more of the sequences (often only one) is a New SP-pattern and the rest are Old, and a ‘good’ SP-multiple-alignment is one in which the New information may be encoded economically in terms of the Old information, as described in [2, Section 2.5] and in [3, Section 4.1].

Examples of SP-multiple-alignments created by the SP Computer Model are shown in Figs A3 and A4.

The first of these shows how the parsing of a sentence may be modelled within the SP-multiple-alignment framework. In

```

0          t w o          k i t t e n          s          p l a y          0
  | | |          | | | | | | | | |          | | | | |
1          | | |          N r 5 k i t t e n #N r |          | | | | |          1
  | | |          | | |          #N r s #N          | | | | |          2
2          | | |          N N p N r          #N r s #N          | | | | |          2
  | | |          | | |          | | |          | | | | |          3
3          D D p 4 t w o #D | | |          | | | | |          3
  | | |          | | |          | | |          | | | | |          4
4          N P D          #D N | | |          #N #N P          | | | | |          4
  | | |          | | |          | | |          | | | | |          5
5          | | |          | | |          | | |          | | | | |          5
  | | |          | | |          | | |          | | | | |          6
6          | | |          | | |          | | |          | | | | |          6
  | | |          | | |          | | |          | | | | |          7
7 S Num ; NP          | | |          #N P V |          #V #S 7
  | | |          | | |          | | |          | | | | |          8
8 Num PL ;          N p          V p          | | | | |          8

```

FIGURE A3. The best SP-multiple-alignment created by the SP Computer Model with a store of Old SP-patterns like those in rows 1–8 (representing grammatical structures, including words) and a New SP-pattern (representing a sentence to be parsed) shown in row 0. Adapted with permission from Fig. 1 in [14].

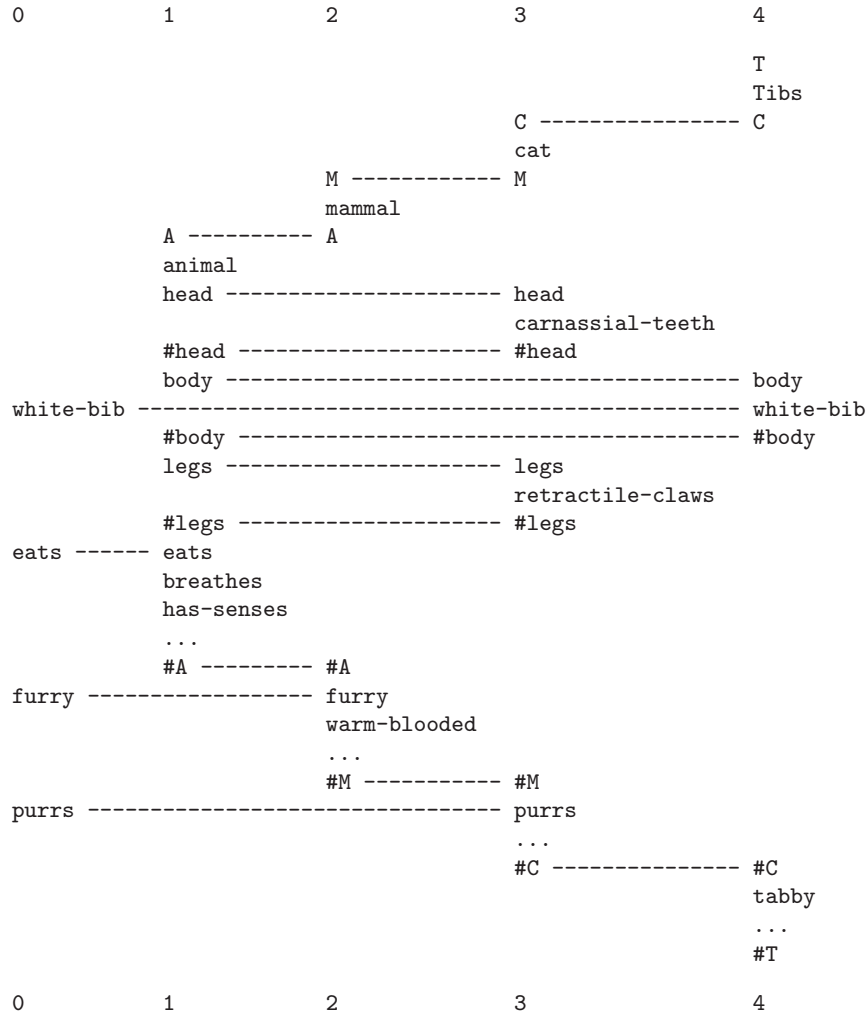


FIGURE A4. The best SP-multiple-alignment found by the SP model, with the New SP-pattern ‘white-bib eats furry purrs’ shown in column 1, and a set of Old SP-patterns representing different categories of animal and their attributes shown in columns 1–4. Reproduced with permission from Fig. 15 in [3].

the figure, row 0 shows a New SP-pattern representing the sentence to be parsed while rows 1–8 show Old SP-patterns, one per row, representing grammatical structures including words. Row 8 shows the grammatical dependency between the plural subject of the sentence (marked with ‘Np’) and the plural main verb (marked with ‘Vp’).

By contrast with the SP-multiple-alignment shown in Fig. A3, the SP-multiple-alignment in Fig. A4 shows SP-patterns in columns, one SP-pattern per column, with alignments between SP-symbols shown in rows. The two ways of representing SP-multiple-alignments are entirely equivalent. The choice normally depends on what fits best on the page. Column 0 in Fig. A4 shows a New SP-pattern representing something to be recognized while columns 1–4 show Old SP-patterns, one per row, representing categories of entity.

By convention, the New SP-pattern is always shown in column (or row) 0 and Old SP-patterns are shown in other columns (or rows). In some applications there is more than 1 New SP-pattern in column (row) 0.

A.5. The SP71 Computer Model

The latest version of the SP Computer Model is SP71. Instructions for obtaining the source code are given in Appendix B.

There is no comprehensive description of SP71, but SP70, a very similar and slightly earlier version of the programme is described quite fully in [2]. More detailed citations are given in the three subsections that follow.

At the level of detail that will be considered, the description of SP70 in [2] is an accurate description of SP71. In what follows, it will be regarded as a description of SP71.

A.5.1. Finding good full and partial matches between SP-patterns in SP71

In the foundations of the SP Computer Model is a process for finding good full and partial matches between two SP-patterns (sequences of symbols) that works well even when there are substantial differences between the two SP-patterns. This is described quite fully in [2, Appendix A].

This technique is similar in its effects to standard ‘dynamic programming’ methods for comparing sequences [39, 40] but with these main advantages [2, Section 3.10.3.1]:

- It can match arbitrarily long sequences without excessive demands on memory.
- For any two sequences, it can find a set of alternative matches (each with a measure of how good it is) instead of a single ‘best’ match.
- The ‘depth’ or thoroughness of the searching can be controlled by parameters.

A.5.2. Building SP-multiple alignments in SP71

The way in which SP-multiple alignments are built in SP71 is described in [2, Sections 3.9 and 3.10]. Slightly confusingly, the main subject of these sections is SP61, but this is merely a part of the SP Computer Model that is concerned with the building of SP-multiple alignments.

As with multiple sequence alignments in bioinformatics, and as with the process of finding good full and partial matches between two SP-patterns (Section A.5.1), heuristic techniques are used in the building of SP-multiple-alignments in the SP System: searching for good SP-multiple-alignments in stages with a weeding out of the lower-scoring SP-multiple-alignments at the end of each stage.

A.5.3. Unsupervised learning in SP71

Unsupervised learning in SP71 is described in [2, Sections 3.9 and 9.2], with relevant pseudocode in [2, Figures 9.1 and 9.6]. The aim here is, for a given set of New SP-patterns, to create one or two *grammars*—meaning collections of Old SP-patterns—that are effective at encoding the given set of New SP-patterns in an economical manner.

The building of SP-multiple-alignments is an integral part of unsupervised learning in the SP System. It provides a means of creating Old SP-patterns via the splitting or amalgamation of pre-existing Old SP-patterns, and via the direct incorporation of New SP-patterns. And it provides a means of evaluating candidate grammars in terms of their effectiveness at encoding the given set of New SP-patterns in an economical manner.

As with the building of SP-multiple alignments, and the process of finding good full and partial matches between SP-patterns, the creation of good grammars requires heuristic search through the space of alternative grammars: creating grammars in stages and discarding low-scoring grammars at the end of each stage.

A.5.4. Varying the thoroughness of heuristic search

A useful feature of SP71 is that, with all three of the main components (Appendix A.5.1, Appendix A.5.2 and Appendix A.5.3), the thoroughness of the searches may be varied by varying the amount of memory space that is available for storing intermediate results. In effect, this controls the amount of backtracking that can be done and thereby controls the chances of escaping from local peaks in the search space.

A.6. SP-Neural

Abstract concepts in the SP theory map quite neatly into groupings of neurons and their interconnections in a version of the theory called *SP-Neural* [29]: SP-symbols are realized as *neural symbols* (single neurons or small clusters of neurons) and SP-patterns are realized as *pattern assemblies*. Although pattern assemblies in SP-Neural are quite similar to *cell assemblies* as described by Donald Hebb [41], unsupervised learning in the SP System (including what is envisaged in the development of SP-Neural) is quite different from ‘Hebbian’ learning [41, p. 62] or versions of Hebbian learning that are popular in ‘deep learning in artificial neural networks’ [15]. The key differences are described in [29, Section 10.5].

A.7. Distinctive features and advantages of the SP System

Distinctive features and advantages of the SP System are described in [4]. In particular, Section V of that paper describes 13 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 14th problem with deep learning—‘catastrophic forgetting’—meaning the way in which new learning in a deep learning system wipes out old memories.⁸

There is further discussion in [43] of weaknesses in deep learning and how, in the SP System, those weaknesses may be overcome.

Key strengths of the SP System, which owe much to the central role in the system of the SP-multiple-alignment concept, are in three inter-related features of the system:

- *Versatility in aspects of intelligence.* The SP System has strengths in several aspects of intelligence, including unsupervised learning, pattern recognition, planning, problem solving, and several kinds of reasoning, and there are reasons to think that SP-multiple-alignment may prove to be the key to general, human-like artificial intelligence. ([28, Appendix B.2], [44, Section 4]).

⁸A solution has been proposed in [42] but it appears to be partial, and it is unlikely to be satisfactory in the long run.

- *Versatility in the representation of knowledge.* The SP System has been shown to be a versatile framework for the representation of diverse kinds of knowledge, and there are reasons to think that it may in principle accommodate any kind of knowledge ([28, Appendix B.1], [44, Section 5]).
- *Seamless integration of diverse kinds of knowledge and diverse aspects of intelligence, in any combination.* Since the SP System's strengths in the representation of diverse kinds of knowledge and in diverse aspects of intelligence all flow from a single coherent source—the SP-multiple-alignment framework—there is clear potential for their seamless integration in any combination (as defined at the end of Section 2) [28, Appendix B.3]. As was mentioned in Section 2, that kind of seamless integration appears to be essential in any system that aspires to human-like fluidity, versatility, and adaptability in intelligence.

A.8. Potential Benefits and Applications of the SP System

As noted in Section 3, the SP System has yielded several potential benefits and applications, described in a book and in several peer-reviewed papers:

- The book, *Unifying Computing and Cognition* [2] describes strengths of the SP System in several areas, mainly in AI, with corresponding potential in applications that require human-like intelligence.
- The paper 'The SP Theory of Intelligence: benefits and applications' [5] describes several potential benefits and applications of the SP System, including potential for: an overall simplification of computing systems, including software; applications in the processing of natural language; benefits in software engineering; benefits in bioinformatics; and several more.
- The paper 'Big data and the SP Theory of Intelligence' [6] describes how the SP System may help solve nine significant problems with big data.
- The paper 'Autonomous robots and the SP Theory of Intelligence' [7] describes how the SP System may help in the development of human-like intelligence in autonomous robots.
- The paper 'Application of the SP Theory of Intelligence to the understanding of natural vision and the development of computer vision' [21] describes, as its title suggests, how the SP System may help in the understanding of human vision and in the development of computer vision.
- The paper 'Towards an intelligent database system founded on the SP Theory of Computing and cognition' [14] describes how the SP System may function as a database system that, in a relatively streamlined manner, can accommodate a wider variety of kinds of knowledge than a conventional DBMS, with potential for diverse aspects of intelligence.
- The paper 'Medical diagnosis as pattern recognition in a framework of information compression by multiple-alignment, unification and search' [22] describes how the SP System may serve as a repository for medical knowledge and as a means of assisting medical practitioners in medical diagnosis.
- The paper 'Commonsense reasoning, commonsense knowledge, and the SP Theory of Intelligence' [45] describes how the SP System may help solve several of the problems in modelling commonsense reasoning and commonsense knowledge, described by Ernest Davis and Gary Marcus in [46].
- The draft report 'Software engineering and the SP Theory of Intelligence' [28] describes how the SP System may be applied in software engineering, and the potential benefits.

APPENDIX B. SOURCE CODE AND WINDOWS EXECUTABLE CODE FOR THE SP71 COMPUTER MODEL

The latest version of the SP Computer Model is SP71. At the time of writing, the source code, written in C++ with many comments, may be obtained via instructions under the headings 'Source code' and 'Archiving', near the bottom of bit.ly/1mSs5XT. Likewise for the Windows executable code. More specifically, these things may be obtained:

- In the file SP71.zip (bit.ly/1OEVPDw).
- The source code is in 'Ancillary files' with 'The SP Theory of Intelligence: an overview' under www.arxiv.org/abs/1306.3888. Apart from SP71.exe, all the files should be treated as plain text files to be opened with Wordpad, WinEdt, or the like.
- From a digital archive with: the National Museum of Computing, Block H, Bletchley Park, Milton Keynes, MK3 6EB, UK; *Phone*: +44 (0)1908 374708; *Email*: operations@tnmoc.org.
- From a digital archive with: the Library and Archives Service, Bangor University, Bangor, Gwynedd, LL57 2DG, UK; *Phone*: +44 (0) 1248 382981; *Email*: library@bangor.ac.uk.