To appear in *Computational Learning and Probabilistic Reasoning*, A. Gammerman (Ed.), New York: Wiley. This is a revised version of ‘Learning and reasoning as compression by multiple alignment and unification’ presented at Applied Decision Technologies ‘95, Brunel University, April 1995. A copy of that paper is in the Proceedings for Stream 1 (Computational Learning and Probabilistic Reasoning), pp. 223-236.

LEARNING AND REASONING AS INFORMATION COMPRESSION BY MULTIPLE ALIGNMENT, UNIFICATION AND SEARCH

J Gerard Wolff

School of Electronic Engineering and Computer Systems, University of Wales, Dean Street, Bangor, LL57 1UT, UK. Telephone +44 248 382691. E-mail: gerry@sees.bangor.ac.uk. Fax: +44 248 361429. Web: http://www.sees.bangor.ac.uk/~gerry.

September 1995

1 INTRODUCTION

This article presents the tentative idea that ‘multiple alignment’ in a sense which is close to the use of that term in bio-informatics, together with the full or partial merging or ‘unification’[1] of patterns, and a process of ‘search’, is a framework within which learning and reasoning may be integrated.

This thinking is part of a programme of research aiming to develop the ‘SP’ conjecture (‘computing as compression’) that *all kinds of computing and formal reasoning may usefully be understood as information compression by pattern matching, unification and search* (PMUS), and to develop a ‘new generation’ computing system based on the theory [19, 20, 21, 24].

Learning and reasoning are both large subjects. In the space of one short article it is not possible to do more than present a few examples to suggest how these things may be seen in terms of multiple alignment, unification and search. Relevant issues will be discussed more fully elsewhere.

Research on ‘inductive logic programming’ (ILP) is also concerned with learning and reasoning but the focus is different from the SP programme. In ILP (see, for example, [12]) the emphasis is on (supervised) learning within a framework of logic, whereas the SP programme seeks to *integrate* (unsupervised) learning and reasoning (and other aspects of computing) within a more

1. The term *unification* is used in this article to mean a simple merging of multiple instances of any pattern to make one. This idea is related to, but simpler than, the concept of ‘unification’ as it is used in logic.

general framework.

In this article, the term *pattern* is used to mean any sequence of atomic symbols including subsequences within a larger sequence where the symbols in the subsequence are not necessarily contiguous within the larger sequence.

2 MULTIPLE ALIGNMENT PROBLEMS

The term 'multiple alignment' is normally associated with the computational analysis of DNA sequences or sequences of amino acid residues as part of the process of elucidating the structure, functions or evolution of the corresponding molecules.

The general idea is to examine two or more sequences to find one or more alignments of matching bases or amino acid residues which are, in some sense, optimal. An example is presented in Fig. 1.

Please insert Fig. 1 about here

Fig. 1. An alignment amongst five DNA sequences (adapted from Fig. 6 in [13], with permission from Oxford University Press).

2.1 What is an 'Optimal' Alignment?

Intuitively, an 'optimal' or 'good' alignment amongst two or more sequences of symbols is one which is relatively long, involves a relatively large number of sequences, has few gaps (or no gaps at all) and, where there are gaps, these should be relatively short.

These kinds of measures are used directly in many studies of multiple alignment. But it seems that our intuitions in this area may also be formalised in terms of probabilities and compression. These two kinds of analysis are complementary and not in conflict - they are two sides of the information theory coin. Recent studies in this tradition include [1, 2, 4, 16, 22].

2.2 Searching for Optimal Alignments

In this area of research, it is widely recognised that, when searching for good alignments between two or more sequences, the space of possible alignments is normally too big to be examined exhaustively. For realistically large examples it is necessary to use 'heuristic' techniques (eg 'hill climbing', 'beam search') which may be described generically as 'metrics-guided search'. Existing methods are reviewed in [3, 4, 7, 15]. Also relevant is the method for finding alignments between two sequences which is described in [22].

Given the variety of techniques which already exist to find multiple alignments, one might expect to find one that could be incorporated in a model of learning, reasoning and other aspects of computing as discussed in this and other publications in this research programme. However, an examination of the literature has shown that existing techniques do not precisely meet the needs of the SP programme. Given the differences in objectives between biochemistry and the SP programme, this is not very surprising.

Work is now proceeding on a new search method within a new SP model. The new model is not yet sufficiently mature or robust to demonstrate all the points made in this article. However, it is good enough to solve the geometric analogy problem described in Section 5.7 and other simple examples.

The new model 'broadcasts' symbols systematically to make yes-no matches throughout the set of sequences (in this respect, it is similar to SP20 [20]). The new model then builds alignments by pairwise conjunction of 'hits' and groups of hits, guided at all stages by how much compression each alignment may achieve. In accordance with the principles of metrics-guided search, low scoring alignments are eliminated at each stage.

3 UNIFICATIONS AND TAGS

In the context of biochemistry, multiple alignment means pattern matching and search but it does not normally include unification of patterns, the 'U' of 'PMUS'.

In the context of 'computing as compression', the merging or unification of two or more matching patterns to make one is the step which realises compression. A full discussion is outside the scope of this article. It is pertinent here only to note a few points relating to later sections of the article.

If subsequences of one or more patterns are unified, non-redundant information about the *locations* of the deleted patterns is lost unless steps are taken to preserve it.

The most widely used technique for preserving this non-redundant information about locations is to insert a short 'identifier', 'code' or 'tag' wherever a pattern has been deleted. Of course, the tags must be shorter than the patterns they replace to achieve a net compression. Tags may be newly-minted patterns which replace deleted patterns. It also possible to delete only part of a redundant pattern and leave the residue of the pattern as a tag to mark the position. In either case, it seems necessary to include with the tag some kind of termination or scope marker, as described briefly later in this article.

For the purposes of the SP programme, this simple idea of marking the position of deleted patterns using tags has the attraction that pattern matching and unification can be applied to tags themselves, just like any other information. This means that PMUS mechanisms can be used for de-referencing tags and de-compressing information as well as for the original compression of information (see [19], Section 5.2]).

4 LEARNING AS MULTIPLE ALIGNMENT WITH UNIFICATION

In this discussion, 'learning' will be taken to mean the kind of unsupervised assimilation and organisation of knowledge which appears to be required for the learning of a first language by children or the 'discovery', 'inference' or 'abstraction' of a grammar from a sample of an artificial or natural language. It seems that lessons learned in these areas may be applied to tasks like rule induction for expert systems, 'data mining' and so on.

The language learning task may be analysed into (at least) the following sub-tasks:

- Learning segmental (conjunctive) structure in unsegmented samples (eg words, phrases and sentences).
- Learning contextually-equivalent disjunctive groupings (eg word classes - nouns, verbs, adjectives etc - and similar groupings of other linguistic structures).
- Generalisation of grammatical rules and the correction of overgeneralisations.
- Learning semantic structures and their associations with syntactic forms.

The SNPR model [24, 25], developed in the tradition of 'Minimum Length Encoding' (MLE, [5, 14]), is an unsupervised model of language learning which performs the first three of these tasks recursively to build hierarchical, context-free phrase-structure grammars (CF-PSGs) which may have conjunctive and disjunctive groupings at any level. The model is capable of discovering plausible grammars from unsegmented samples of artificial languages, forming generalisations and plausible corrections for overgeneralisations without correction by an external 'teacher' or the provision of negative samples or the 'grading' of samples in any way (cf. [11]). The MK10 model (see [25]), which is a sub-set of the SNPR model dedicated to the learning of segmental structures, is capable of discovering word structure in unsegmented samples of English.

Notwithstanding the known shortcomings of CF-PSGs for representing the organisation of natural languages, the SNPR model exhibits several of the features of language learning which have been observed in empirical studies of children [25]. These are by-products of the organisation of the model, not *ad hoc* features which have been bolted on.

Although the SNPR and MK10 models are largely applications of PMUS mechanisms, they were developed with objectives which were different from the SP programme and it is not altogether surprising that neither of them precisely meet current requirements. However, it is now possible to see in outline how these models may be re-organised to meet these broader objectives. More specifically, the first two of the tasks identified above may now be seen in terms of multiple alignment and unification.

4.1 Learning Segmental Structure

Fig. 2 shows an alignment amongst a set of sentences generated by a simple grammar. Of all the possible alignments of these sentences, the one shown in the figure appears to be the best. This judgement is based on the compression that may be achieved by the unification of matching sequences of letters within the alignment.

Please insert Fig. 2 about here

Fig. 2. An alignment amongst some sentences

The point of interest here is that, in the alignment, the patterns which we conventionally recognise as 'words' stand out as coherent subsequences. The words in the sentences will fall out as discrete patterns when matching subsequences within the alignment are unified. The words are picked out despite the fact that, in the sentences, there are no explicit markers showing where each word ends and the next begins.

It is not hard to see that the same kind of process operating on sequences of symbols for grammatical categories may serve to identify segments larger than words such as phrases and sentences (cf. [26]).

Of course, the example is relatively simple and is artificial in the sense that, in a typical sample of natural language, it is unlikely that one would find a set of inter-related sentences neatly presented as shown. However, the principles illustrated by this example appear to be general and should apply even in the more haphazard environment of a natural language. Words and other segments may be identified from samples of language containing fragments smaller than sentences. For example, it only requires three fragments, 'r u n s s l o w l y', 'r u n s f a s t', 'w a l k s f a s t' for one to make a preliminary judgement that ‘r u n s’, ‘f a s t’ and ‘w a l k s’ are discrete segments.

4.2 Learning Disjunctive Classes

A key idea in ‘distributional’ or ‘structural’ linguistics (see, for example, [8]) was that distributionally-equivalent classes of structure might be identified on the basis of shared context. For example, the words ‘accountant’ and ‘clerk’ may be placed in the same group (normally known as ‘nouns’) because they may appear in the same position in sentences like:

The ... remembered the tax.

Distributional linguistics relied mainly on judgements by native speakers to determine these privileges and equivalencies. But if one attempts to discover these groupings by any kind of automatic process, there is a problem: it is rare to find pairs of sentences which are identical except for two words in the one position.

The SNPR model solves this problem in the way it combines the search for disjunctive groups with the building of progressively larger segmental structures. It is not necessary to use whole sentences for a preliminary classification of words or other segments: ‘accountant’ and ‘clerk’ may be assigned to the same class because they both occur in the context "the ..." - and pairs of fragments like that are much more common than the required pairs of sentences. When labels for classes have been introduced, they may themselves provide the contexts from which other classes may be derived. And because labels for classes are more frequent than their constituent structures, it becomes relatively easy to find those shared contexts.

In Fig. 2, ‘P e t e r’ and ‘J a n e’ share the same set of contexts and may thus be classified together syntactically as ‘nouns’. In a similar way, ‘r u n s’ and ‘w a l k s’ may be classified as ‘verbs’ and ‘f a s t’ and ‘s l o w l y’ may be classified as ‘adverbs’ or ‘qualifiers’.

There is a natural affinity between the kinds of classification just described and ‘tags’ of the kind described briefly in Section 3: word classes and categories of that kind are based on shared contexts and tags provide the means of marking the contexts of unified patterns. Thus the alignments shown in Fig. 2 may be compressed into the form shown in Fig. 3 by the unification of matching patterns and the use of word class symbols as tags to mark the contexts in which those patterns belong.

Please insert Fig. 3 about here

Fig. 3. Unified patterns from the alignment in Fig. 2, with word class symbols as tags.

In Fig. 3, each unified pattern is terminated by a semi-colon (‘;’) and instances of the same symbol appear in the sequence ‘N ; V ; Q ; ;’ (at the bottom of the figure) which records the sequence of word classes in the sentences shown in Fig. 2. This symbol has been introduced to ensure that the ‘correct’ alignments and unifications are made when the tags are de-referenced. An example showing how tags may be de-referenced by alignment and unification is presented in Section 5.6, below.

In effect, the set of patterns shown in Fig. 3 is a simple grammar corresponding to the set of sentences in Fig. 2. The sequence of tags at the bottom of Fig. 3 represents the structure of the sentence at a relatively abstract level and the other sequences provide lower-level details of word classes and individual words.

5 REASONING AS MULTIPLE ALIGNMENT WITH UNIFICATION

In this article, 'reasoning' is taken to mean deductive reasoning as discussed in standard treatments of logic (eg [6]) and also probabilistic forms of reasoning - abductive, inductive and analogical reasoning. A working hypothesis is that deduction may be cast in a probabilistic frame in which probability values are extreme (close to 0 or 1).

5.1 Recognition and Reasoning

A prominent feature of human (and animal) perception is the ability to recognise a pattern from a sub-set of its features: we may recognise something as a car even though one end is obscured by a building and another part is obscured by a tree; we may recognise a piece of music from hearing only a part.

This kind of recognition may be seen as a form of reasoning: from the observed features we reason that the unobserved features are 'really' there. It is tempting to believe that all forms of human reasoning are founded on this ability to recognise a whole pattern from a sub-set of its parts.

This kind of recognition/reasoning may also be seen in terms of the alignment and unification of patterns: sensory data is a pattern to be aligned and unified with one or more patterns in memory.

5.2 *Modus Ponens*

If something is a mammal then that something is warm blooded.
Tibby is a mammal.
Therefore, Tibby is warm blooded.

In logical notation, this kind of *modus ponens* reasoning may be expressed as:

$\forall x$: mammal(x) $\supset$ warm-blooded(x)
mammal(Tibby)
$\therefore$ warm-blooded(Tibby).

A strict interpretation of the implication symbol ('$\supset$') as 'material implication' is outside the scope of these proposals because the concept of material implication includes a concept of negation and no serious attempt has yet been made to interpret negation in the proposed framework. A 'naturalistic' concept of implication is intended here, close to the everyday meaning of the word.

Fig. 4 shows how this kind of reasoning may be re-cast in terms of the alignment and unification of patterns. The first pattern expresses the idea that there is something whose name is 'Tibby' which is a mammal. The second pattern expresses the association between being a mammal, having warm blood, being furry and other characteristics of mammals. There is an unfilled 'slot' for the name of an individual mammal.

Please insert Fig. 4 about here

Fig. 4. *Modus ponens* reasoning by alignment and unification.

The two patterns may be unified as shown. The unified pattern may be read as "Tibby is a mammal and, as such, is warm-blooded, furry and so on." This contains the required conclusion together with some other conclusions and a re-statement of the 'premises'.

Any unification like this needs to be 'implicit' or 'transitory', otherwise the pattern which describes mammals in general, and which, presumably, needs to be preserved, would quickly lose its generality.

5.2.1 Modelling Named Variables and the Universal Quantifier

In [24], Chapter 5, it was suggested that, via some manipulation of structures similar to the casting of logical structures into clausal form, the universal quantifier might be modelled in an SP system by a variable like an unnamed variable in Prolog (represented in both systems by the underscore symbol '_'). The effect of a named variable might be achieved by matching and unification applied to the combination of an unnamed variable with a symbol acting as a name, eg (X _).

More recent thinking has led to the tentative view that there is no need for an unnamed variable with special properties and that an equivalent effect may be achieved by the use of a symbol like ';' to mark the scope of patterns.

In each of the two patterns in Fig. 4, the symbol ';' serves to mark the scope of the 'N' field and it is also a termination marker for each pattern. Apart from its usage as a scope marker, the ';' symbol has no special properties or status and enters into matching and unification like any other symbol.

In the example, the sub-sequence 'N ;' behaves like a named variable (eg 'x' in logical notation). The effect of matching and unification is to create the sub-sequence 'N Tibby ;' which may be seen as an 'assignment' of 'Tibby' to the 'named variable'.

'Quantification' in this example may be seen as 'universal' because the same inference would be made for any pattern of the form 'N ... ; mammal ;'.

Further discussion of the way in which the concept of a variable and types for variables may be modelled in the proposed framework may be found in [16, 17, 18].

5.3 *Modus Tollens*

The *modus tollens* form of reasoning is similar to *modus ponens* except that it uses negation:

$$\forall x\text{: } \mathrm{p}(x) \supset \mathrm{q}(x)$$
$$\neg \mathrm{q}(\mathrm{a})$$
$$\therefore \neg \mathrm{p}(\mathrm{a}).$$

Unless negation is to be accommodated with a closed-world assumption in the manner of 'negation as failure' (with the associated pitfalls), there appears to be a need for a special negation operator within the proposed framework of compression by PMUS. As noted above, this has not yet been attempted.

As suggested previously [23, 19], the case for including negation as a primitive within the SP framework is strengthened by the observation that negation can be a powerful aid to compression: it is more economical to encode this week's shopping list as 'last week's, *not* soap' than to write it out in full.

Within a framework of alignment and unification of patterns, we may suppose that a pattern which has been 'negated' may be matched and unified like any other pattern but that any resulting 'inferences' are negated. These are things to be explored in the future.

5.4 Abduction

Instead of inferring that something is warm-blooded because it is a mammal (Section 5.2), we may (with some risk) infer abductively that something is a mammal because it is warm-blooded. In this case, of course, there is a good chance that the creature might be a bird. Although abduction is risky, it is a prominent feature of everyday reasoning and can be very useful.

Fig. 5 shows how this kind of reasoning may be seen as alignment and unification. The pattern 'N Chirpy ; warm-blooded ;' would unify equally well with a pattern specifying the features of a bird and the corresponding probabilities (that Chirpy is a mammal or a bird) should be approximately even. However, additional information, eg that Chirpy has feathers or knowledge about typical pet names for birds, would tilt the probabilities in favour of Chirpy being a bird. In general, there is a probability associated with each possible alignment of patterns and this probability depends on the range of patterns which are available for alignment.

Please insert Fig. 5 about here

Fig. 5. Abductive reasoning by alignment and unification.

5.5 Chains of Reasoning

A familiar feature of everyday reasoning and of systems for automatic reasoning (eg Prolog) is the formation of 'chains' of two or more inferences, each one except the first depending on those that precede it.

Fig. 6 shows how this kind of chaining may be seen in terms of multiple alignment. Susan has a cold and John has been sneezing. Knowing that a cold is caused by a virus and that sneezing spreads viruses we may infer that Susan caught her cold from John.

Please insert Fig. 6 about here

Fig. 6. Multiple alignment in a chain of inference. The status of each pattern in the alignment is shown on the right.

5.6 Inductive Reasoning

Socrates is human and is mortal.
Xanthippe is human and is mortal.
Sappho is human and is mortal.
Therefore probably all humans are mortal.

Inductive reasoning, as illustrated in this example from [6], is related to learning as discussed in Section 4. The top part of Fig. 7 shows how the meanings of the first three sentences may be represented as patterns and it shows alignments of symbols within those patterns.

Please insert Fig. 7 about here

Fig. 7. Extraction of a 'chunk' by multiple alignment and unification.

The sub-sequence 'human mortal ;' is a repeating 'chunk' of information which may be extracted in much the same manner as words or other segments at the syntactic level (cf. Section 4). At the bottom of Fig. 7, this unified chunk is shown together with modified versions of the three original patterns. In each modified pattern, 'N' and ';' have been placed around the 'variable' parts of those patterns. The same two symbols are attached to the 'chunk' containing the invariant pattern 'human mortal'.

Although 'human mortal' has been deleted from each of the three original patterns, it should be clear that alignment and unification along the lines described in Section 5.2 can restore this chunk to each context whence it came.

Compression has been achieved in Fig. 7 because there are fewer characters in the unified patterns than in the originals. Normally, one would compute optimal compression taking account of the probabilities of symbols. But if compression is to be achieved *exclusively* by PMUS, it seems more appropriate to use simpler and more direct measures in terms of the numbers of characters or bits used in patterns.

5.6.1 Inductive Generalisation

How do we reach the inductive generalisation that "probably all humans are mortal"? Within the framework of multiple alignment, unification and search, it seems that this may be achieved in much the same manner as deductive or abductive inferences (Sections 5.2 and 5.4, above).

If, for example, we know that someone called 'David' is human, we may infer that he is mortal because of the regular association between humanness and mortality. Fig. 8 shows how this may be achieved by alignment and unification of relevant patterns. Since this kind of inference may be made for any entity which is described as human, and since all these inferences would have a high probability, we may conclude that "probably all humans are mortal."

Please insert Fig. 8 about here

Fig. 8. An inference that someone other than Socrates, Xanthippe and Sappho is mortal.

Why should we not also conclude that "probably all mortals are human"? As was indicated in Section 5.4, the probability associated with any alignment depends on the range of patterns in the database. If the database recorded only the association between humanness and mortality then it seems that the conclusion that all mortals are human would be reasonable. But in a database containing a more realistic variety of information, there would be many other associations including, for example, the facts that cows are mortal, birds are mortal, fish are mortal and so on. In this case, the probability associated with the inference that any given mortal being was human would be relatively low.

5.7 Analogical Reasoning

It seems that analogical reasoning, as exemplified by geometric analogies of the kind used in puzzles and IQ tests, may be understood in terms of MLE [9, 10]. This article makes the more specific proposal that this kind of reasoning, or aspects of it, may be understood in terms of multiple alignment and unification.

In order to interpret geometric analogies in these terms, it seems necessary to translate the geometrical forms into sequences of symbols (‘patterns’) like those presented elsewhere in this article. In the example shown in Fig. 9, item A may be translated as ‘small star left-of large square’, B may be translated as ‘small star above large square’ and so on.

Please insert Fig. 9 about here

Fig. 9. A geometric analogy problem.

This translation is a significant problem in itself, well outside the scope of this article except for one aspect which deserves to be noted: as discussed elsewhere [10], the case for an MLE interpretation of geometric analogies requires that textual descriptions are in the ‘right’ form. For example, it makes a difference whether A in Fig. 9 is described as ‘small star left-of large square’ or as ‘large square right-of small star’. To be more precise, there must be consistency within the scope of one problem in the way geometric forms are described.

The ‘correct’ solution to the problem shown in Fig. 9 is clearly item 3. This solution is shown in Fig. 10 as an alignment amongst the textual descriptions of A, B, C and 3.

Please insert Fig. 10 about here

Fig. 10. Multiple alignment in a geometric analogy problem.

Unification with an example like this would require the use of tags along the lines described in Section 5.7. A full discussion is not appropriate here. However, it should be reasonably clear that the alignment which is shown is better than the best alignment that one might obtain with any of items 1, 2 or 4 in Fig. 9 and that unification of the alignment in Fig. 10 should give more compression than any of the alternatives.

6 CONCLUSION

In this article I have tried to show the potential of multiple alignment with unification and search as a framework within which learning and reasoning may be integrated. Within the space available it has been possible only to sketch out the possibilities. These ideas are still quite tentative and there are clearly many issues and avenues to be explored.

In the light of previous and current work, it appears that multiple alignment, unification and search may also provide a framework for understanding several other aspects of computing including information retrieval and pattern recognition, parsing, the execution of functions, planning and problem solving. Relevant discussion may be found in [16, 17, 18].

The possible integration of diverse kinds of computing, including learning and reasoning, within one simple framework provides a powerful motive for developing these ideas. Apart from potential benefits in terms of theory, there is the prospect of developing a ‘new generation’ computing system with practical advantages compared with conventional systems [19].

ACKNOWLEDGEMENTS

I am grateful to Alex Gammerman for useful discussion and for copies of relevant papers; to Lloyd Allison of the Department of Computer Science, Monash University, for sending me articles about multiple alignment and compression; and to an anonymous referee for constructive comments.

REFERENCES

1 Allison L and Yee C N (1990). Minimum message length encoding and the comparison of macromolecules, *Bulletin of Mathematical Biology*, 52(3), 431-453.

2 Allison L, Wallace C S and Yee C N (1992). Finite-state models in the alignment of macromolecules, *Journal of Molecular Evolution*, 35, 77-89.

3 Barton G J (1990). Protein multiple sequence alignment and flexible pattern matching, *Methods in Enzymology*, 183, 403-428.

4 Chan S C, Wong A K C and Chiu D K Y (1992). A survey of multiple sequence comparison methods, *Bulletin of Mathematical Biology*, 54(4), 563-598.

5 Cook C M and Rosenfeld A (1976). Some experiments in grammatical inference, in *Computer Oriented Learning Processes* (Ed Simon, J C) pp. 157-174, Leyden: Noordhoff.

6 Copi I M (1986). *Introduction to Logic*, New York: MacMillan.

7 Day W H E and McMorris F R (1992). Critical comparison of consensus methods for molecular sequences, *Nucleic Acids Research*, 20 (5), 1093-1099.

8 Fries C C (1952). *The Structure of English*, New York: Harcourt, Brace & World.

9 Gammerman A J (1991). The representation and manipulation of the algorithmic probability measure for problem solving, *Annals of Mathematics and Artificial Intelligence*, 4, 281-300.

10 Gammerman A J (1993). Geometric analogy problems by minimum length encoding, *Proceedings of the 4th Conference of the International Federation of Classification Societies, Paris (IFCS-93)*, August-September.

11 Gold M (1967). Language identification in the limit, *Information and Control*, 10, 447-474.

12 Kietz, J-U and Lübbe M (1994). An efficient subsumption algorithm for Inductive Logic Programming, *Proceedings of the Fourth International Workshop in Inductive Logic Programming (ILP-94)*, 97-105.

13 Roytberg M A (1992). A search for common patterns in many sequences, *Cabios*, 8(1), 57-64.

14 Solomonoff R J, (1964). A formal theory of inductive inference, Parts I and II, *Information and Control*, 7, 1-22 and 224-254.

15 Taylor W R (1988). Pattern matching methods in protein sequence comparison and structure prediction, *Protein Engineering*, 2 (2), 77-86.

16 Wolff, J G (submitted). Computing as compression by multiple alignment, unification and search (1).

17 Wolff, J G (submitted). Computing as compression by multiple alignment, unification and search (2).

18 Wolff, J G (submitted). Multiple alignment and computing.

19 Wolff J G (1995). Computing as compression: an overview of the SP theory and system, *New Generation Computing*, 13, 187-214.

20 Wolff J G (1995). Computing as compression: SP20. To appear in *New Generation Computing*, 13, 215-241

21 Wolff J G (1994). Computing and information compression: a reply, AI Communications, 7(3/4), 203-219.

22 Wolff J G (1994). A scaleable technique for best-match retrieval of sequential information using metrics-guided search, *Journal of Information Studies*, 20(1), 16-28.

23 Wolff J G (1993). Computing, cognition and information compression, *AI Communications*, 6(2), 107-127.

24 Wolff J G (1991). *Towards a Theory of Cognition and Computing*, Chichester: Ellis Horwood.

25 Wolff J G (1988). Learning syntax and meanings through optimization and distributional analysis, in *Categories and Processes in Language Acquisition* (Eds Levy, Y, Schlesinger, I M, and Braine, M D S) pp. 179-215, Hillsdale, N.J.: Lawrence Erlbaum. (Reproduced in Ref. 24, Chapter 2).

26 Wolff J G (1980). Language acquisition and the discovery of phrase structure. *Language & Speech*, 23, 255-269..

Fig. 1

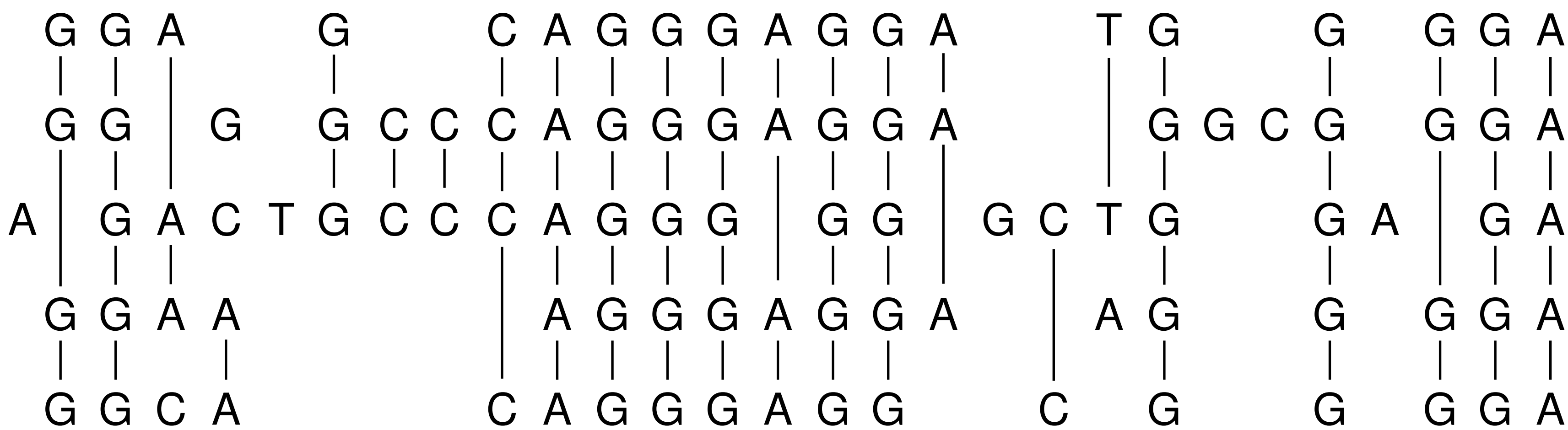
G G A G C A G G G A G G A T G G G G A
G G G G C C C A G G G A G G A G G C G G G A
A G A C T G C C C A G G G G G G C T G G A G A
G G A A A G G G A G G A A G G G G A
G G C A C A G G G A G G C G G G G A

Fig. 2

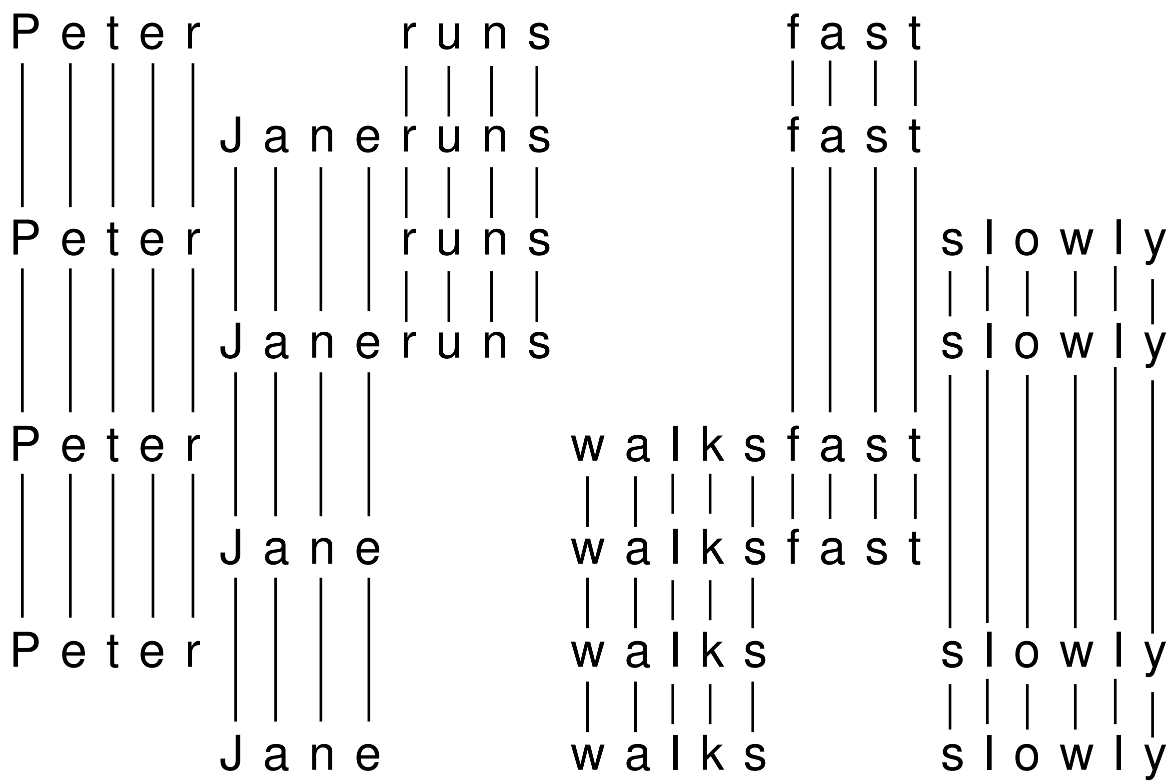
Peter runs fast
Janeruns fast
Peter runs slowly
Janeruns slowly
Peter walksfast
Jane walksfast
Peter walks slowly
Jane walks slowly

Fig. 3

```
N P e t e r ;
N J a n e ;
V r u n s ;
V w a l k s ;
Q f a s t ;
Q s l o w l y ;
N ; V ; Q ; ;
```

Fig. 4

Alignment:

```
N  Tibby  ;  mammal                                ;
|         |    |                                   |
N         ;  mammal  warm-blooded  furry  ...  ;
```

Unification:

N Tibby ; mammal warm-blooded furry ... ;

Fig. 5

Alignment:

```
N  Chirpy  ;                    warm-blooded                ;
|          |                         |                      |
N          ;   mammal   warm-blooded   furry   ...   ;
```

Unification:

N Chirpy ; mammal warm-blooded furry ... ;

Fig. 6

Susan has cold	*observed*
	known association
cold from virus	
	known association
virus from sneezing	
sneezing by John	*observed*

Fig. 7

Alignment:

```
  Socrates  human  mortal  ;
                |       |  |
Xanthippe   human  mortal  ;
                |       |  |
   Sappho   human  mortal  ;
```

Unification:

N Socrates ; ;
N Xanthippe ; ;
N Sappho ; ;
N ; human mortal ;

Fig. 8

Alignment:

```
N  David  ;  human            ;
|         |    |              |
N         ;  human  mortal    ;
```

Unification:

N David ; human mortal ;

Fig. 9

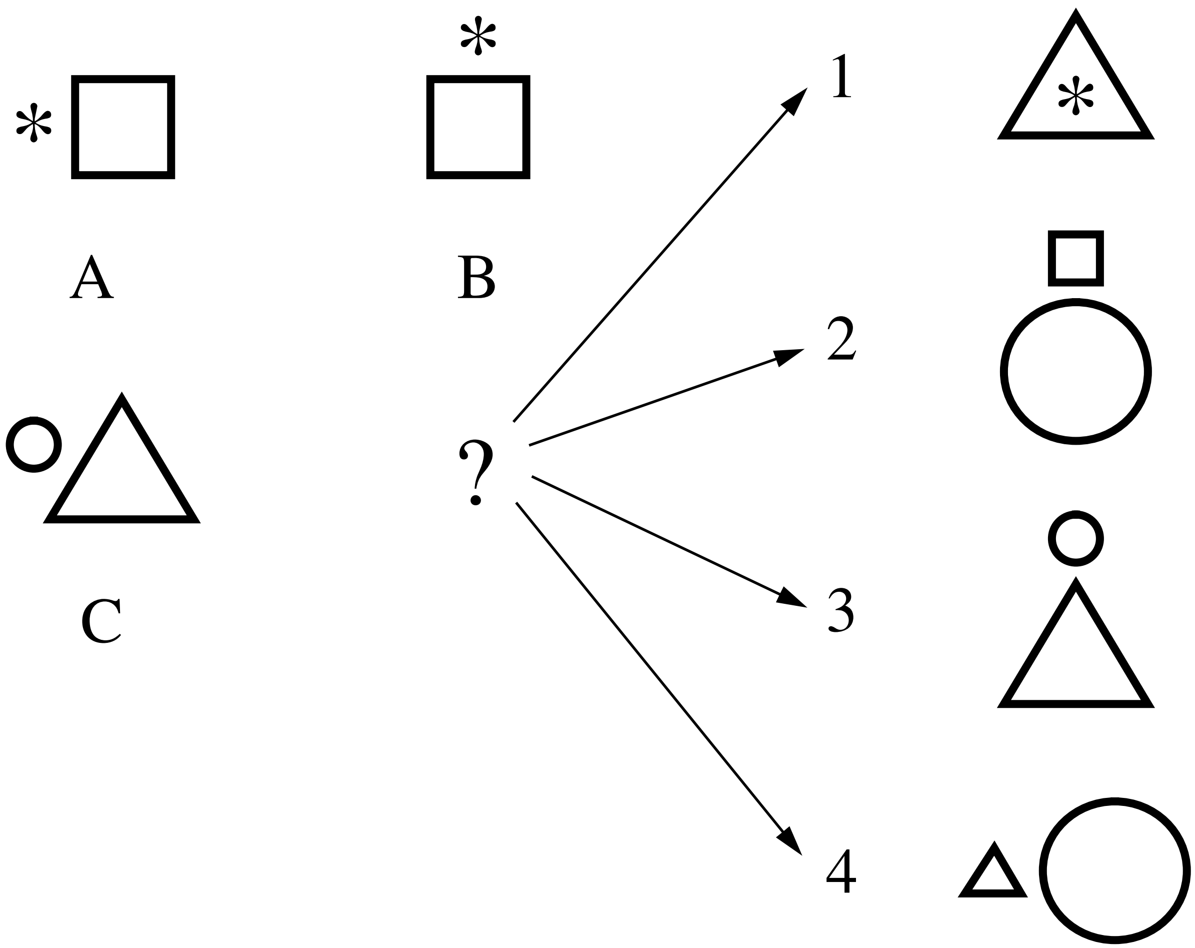

Fig. 10

3	small	circle		above	large	triangle
C	small	circle	left-of		large	triangle
B	small	star		above	large	square
A	small	star	left-of		large	square