Available online at www.sciencedirect.com

ScienceDirect

Artificial Intelligence 171 (2007) 1122–1123

Artificial Intelligence

www.elsevier.com/locate/artint

Book review

J. Gerard Wolff, Unifying Computing and Cognition.

Tom Armstrong

Department of Computer Science and Electrical Engineering, UMBC, 1000 Hilltop Circle, Baltimore, United States

Available online 10 October 2007

In J. Gerard Wolff's latest book "Unifying Computing and Cognition," he proposes to do nothing short of explaining the fundamental nature of cognition and supplanting the Universal Turing Machine as the prevailing theory of computation. If that were not already a tall order, Wolff aims to accomplish both tasks with the same system.

More than three decades ago, Wolff began exploring the connection between compressing text and inferring useful structures. Beginning with the MK10 model [1], an algorithm for segmenting natural language texts into constituent words (the precursor to SEQUITUR [2]), balancing information compression with expressivity has taken center stage in Wolff's theories. Later efforts like SNPR [3], an extension of MK10 from text segmentation to grammar induction and the motivation for GRIDS [4], extended the same theory, but had less empirical success. The most recent incarnation is promoted as the SP theory: the *Simplicity* and *Power* theory. *Simplicity* is driven by compression of information using minimum length encoding (MLE) as the guiding principle. *Power* is the expressivity or "explanatory range" of a theory. He expands the reach of the theory further than in the past:

> "*All kinds of information processing in brains or computers—including 'thinking,' 'reasoning,' 'computing,' and so on—may be understood as information compression by multiple alignment, unification, and search*." (p. 68)

Wolff operationalizes the SP theory by leveraging redundancies found in new information from already known old information to form multiple alignments. The notion of multiple alignments here is borrowed from bioinformatics and unification from logic *vis-à-vis* Prolog, one of Wolff's inspirations. Finding a good multiple alignment for new information is finding a good compression in terms of old information.

Information is generically called a pattern, a sequence of discrete elements in one or more dimensions. Patterns fit into two groups: *old*, a database containing known patterns and frequencies, and *new*, a query containing a novel pattern. Multiple alignments are built from patterns in *old* that unify with subsequences of the query pattern. A query pattern can have many multiple alignments, each denoting a different representation of the query pattern using patterns in *old*. The amount that a multiple alignment compresses a query pattern determines its quality. Multiple alignments are ordered based on quality where higher compression is higher quality.

The main contributions of the book come in the form of two implementations of the SP theory. The first, version SP61, is the basic SP theory with multiple alignment, unification, and search and the second, SP70, extends SP61 with a learning algorithm. The majority of the text focuses on domain specific examples demonstrating the utility of SP61 and justifying that it is a viable theory of computation. The balance of the book connects the SP theory with neural theory and discusses the current limitations of SP70 and how the theory is a viable theory of cognition. The latter feels too disconnected, belonging in a separate book.

The SP theory is equivalent in computational power to a Universal Turing Machine. Wolff argues, though, that while the two theories of computation can achieve the same ends, the underlying first principles are different and those differences are critical. The Universal Turing Machine is not founded upon MLE, while SP theory is wholly

E-mail address: tom.armstrong@umbc.edu.

doi:10.1016/j.artint.2007.10.014

based upon MLE. SP theory has explanatory power as a model of cognition and the Universal Turing Machine does not. Much of the book reads that the means of computing for the same result is of great importance seemingly for its own sake.

After demonstrating that the SP theory is Turing complete, there are too many chapters belaboring the merit that SP61 can perform a specific list of computations. Examples vary from information retrieval from a simple database to nonmonotonic reasoning. Most of the examples only feature one-dimensional sequences. Wolff claims that low-dimensional spaces suffice for much of human intelligence, but does not address the curse of dimensionality for problems that require more. The allure of applying the theory to multiple fields wanes in much the same way interest in Prolog diminishes; it works nicely with small, toy problems, but it does not scale easily, and simple operations are too costly.

The book flourishes, however, in the Natural Language Processing chapter. Language learning and language applications have always been the primary motivations for Wolff's work, and it is clear that SP theory is mainly directed toward these ends. The SP theory offers a nice representation choice for natural language applications and learning algorithms. More intuitive patterns replace rewrite rules. Paring and production of strings in the language are equivalent operations just with different queries (e.g., a query of 'S,' some start symbol, generates strings in the language and 'thecatsleeps' generates a ranked list of parse trees for the sentence). SP theory is facile at representing grammars in the Chomsky hierarchy. The representation should be attractive to proponents of mildly context-sensitive grammar formalisms (e.g., Tree-Adjoining Grammars, Contextual Grammars, Categorial Grammars, etc.). Wolff gives several examples that demonstrate the ease of representing necessary linguistic constructions like recursion, ambiguity, and long-distance dependencies of number and gender. Moreover, the implementation allows for noise and errors in the input. The system appears to be robust with respect to arbitrarily long gaps, substitution, and omission. For example, the multiple alignments SP61 finds for "theecat" given a basic grammar for English rank "the cat" and "these cats" above "a cat."

The language domain introduces the second model, SP70, but the infancy of the ideas leaves the reader guessing. As an unsupervised learning algorithm in the league of other Harris-style constituency approaches like ABL [5] and ADIOS [6], more detail and experimentation are needed to properly evaluate the contribution. But, much like the other modern alignment-based methods, the perennial problems of overgeneralization and recovery from overgeneralization exist, but are not resolved, or even addressed. Wolff contributes some intuition that generalizations are "good in terms of minimum length encoding and these are normally ones that steer a path between generalisations that are, intuitively, 'correct' and others that appear to be 'wrong'." (p. 259)

Wolff has set a lofty goal of convincing too many all at once; the scope of the book is simply too great and he acknowledges the incompleteness of the theory. In spite of these shortcomings, the SP theory should be well received given the proper audiences of natural language processing and language learning—the domains where the theory thrives. Those in the community at the intersection of computational linguistic and grammar induction ought to benefit from the mutually advantageous representation and system defined in the book. Perhaps the freely available source code, elegance of the representation, and expressivity will foster renewed collaborations between the two groups.

References

[1] J.G. Wolff, An algorithm for the segmentation of an artificial language analogue, British J. Psychol. 66 (1975) 79–90.

[2] C.G. Nevill-Manning, I.H. Witten, Identifying hierarchical structure in sequences: A linear-time algorithm, J. Artificial Intelligence Res. 7 (1997) 67.

[3] J.G. Wolff, Language acquisition, data compression and generalization, Language Commun. 2 (1982) 57–89.

[4] P. Langley, S. Stromsten, Learning context-free grammars with a simplicity bias, in: Machine Learning: ECML 2000, 11th European Conference on Machine Learning, Barcelona, Catalonia, Spain, May 31–June 2, 2000, in: Proceedings, vol. 1810, Springer, Berlin, 2000, pp. 220–228.

[5] M. van Zaanen, ABL: Alignment-Based Learning, in: Proceedings of the 18th International Conference on Computational Linguistics (COLING), Saarbrücken, Germany, Association for Computational Linguistics, 2000, pp. 961–967.

[6] Z. Solan, D. Horn, E. Ruppin, S. Edelman, Unsupervised learning of natural languages, Proc. Natl. Acad. Sci. USA 102 (33) (2005) 11629–11634.