Functional Compositionality: a *G.N.U.* Approach *

Shan H. Parfitt

Neural Systems Engineering, Department of Electrical and Electronic Engineering,
Imperial College, Exhibition Rd, London, SW7 2BT, UK.

email: s.parfitt@ic.ac.uk†

Fodor and Pylyshyn's famous 1988 polemic [Fodor and Pylyshyn 1988] on the use of Artificial Neural Networks (ANNs) for natural language modelling prompted a number of researchers, including van Gelder [1989 and 1990], Pollack [1990], Chalmers [1990], Davies [1991], Goschke [1991], Touretzky [1991] and Niklasson [1992] to undertake work which has already gone quite some way to answering the formers' important claims that ANNs are not capable of concatenative compositionality or systematicity. Smolensky [1987] argued successfully, that (in Van Gelder's words), "Connectionist representations can be compositional without being strictly combinatorial in the Classical sense". Van Gelder went on to show that Classical concatenative compositionality is just one form of compositionality, and that one strength of ANNs lies in another, which he calls functional compositionality. Its existence can be demonstrated through an analysis of the hidden representations which form during the training of both feedforward and recurrent multi-layer perceptrons. These representations are a point of particular interest because of their potential to enable the development of effective models of natural language which do not rely upon the combinatorial syntax and traditional, explicitly algorithmic methods of Computational Linguistics. In his well-known 1990 paper, "Finding structure in time", Jeffrey Elman [1990] investigated the ability of a Simple Recurrent Network (SRN) to make predictions about language-like sequences. The present work examines the performance of a weightless equivalent of the SRN (a General Neural Unit, or GNU) [Aleksander 1990 and 1991, Lucy 1991, Ntourntoufis 1991] on the same task. The GNU used consisted of 16 external and 16 internal GRAMS, each external GRAM outputting to one output unit, and each internal GRAM feeding its output back to 1 internal input node. There were, correspondingly, 32 input units (16 external and 16 internal). Each GRAM received input from a random subset, n , of the input nodes, such that the proportion of inputs received by a GRAM from external and internal input nodes was equal [Aleksander 1991]. An experimental corpus was generated to be as similar as possible to that used by Elman. The size of n was varied between $n = 4$ and $n = 22$. The following range of results was obtained.

$n = 4$: correct wordclasses = 15.72%; correct words = 4.4%.

$n = 8$: correct wordclasses = 21.46%; correct words = 7.0%.

$n = 16$: correct wordclasses = 46.63%; correct words = 33.63%.

$n = 20$: correct wordclasses = 71.66%; correct words = 65.67%.

$n = 22$: correct wordclasses = 82.0%; correct words = 77.0%.

If the corpus or language upon which an SRN is trained contains few strings, both an SRN and GNU should be capable of perfect or near-perfect next-word prediction. Elman showed that, for a large corpus, an SRN loses the ability to correctly predict the next word, but attains sufficient generalisation to be able to correctly predict likely next wordclass. It is rather difficult to make a direct comparison between the % correct predictions of the two types of model: in Elman's, since each word is represented by one positive bit, the response can be interpreted as a set of approximate probabilities, whereas in the other, it is not possible to distinguish more than one pattern in the response, because the nature of a weightless system makes distributed patterns preferable. Nevertheless, for larger n , the GNU learns to predict correct next word in the majority of cases, its performance greatly exceeding that of the SRN. For smaller n , it learns to predict next wordclass at well above chance, but does less well than the SRN. The auxiliary representations were passed through a hierarchical clustering program¹. For a similar training set, the nature of the auxiliary representations obtained using a GNU are rather different from those obtained using an SRN, the former appearing to be largely random. However, it is important to note that their random nature did *not* prevent the network from being able to make predictions from the training data. Whilst there was no apparent similarity in the representations for similar tokens, functional compositionality is nevertheless present, as demonstrated by the figures for correct next wordclass prediction. In fact, the results suggest that, in order for functional compositionality to exist, internal representations which are ordered in state-space may not be a fundamental requirement, in weightless systems at least. It is not at present clear, however, whether this would remain the case for increasingly large languages.

*I would like to thank: Igor Aleksander, Jeffrey Elman, Andreas Stolke and members of the Neural Systems Group.

†A technical report describing the work may be obtained at this address.

¹Cluster V. 2.5 by: Yoshiro Miyata (miyata@boulder.colorado.edu), Andreas Stolke (stolcke@icsi.berkeley.edu). Steve Omohundro (om@icsi.berkeley.edu) and Kim Daugherty (kim@gizmo.usc.edu).