Hebbian Constraint on the Resolution of the Homunculus Fallacy Leads to a Network that Searches for Hidden Cause-Effect Relationships

András Lőrincz

Department of Information Systems, Eötvös Loránd University, Pázmány Péter sétány 1/C, Budapest, Hungary 1117

Abstract

We elaborate on a potential resolution of the homunculus fallacy that leads to a minimal and simple auto-associative recurrent *'reconstruction network'* architecture. We insist on Hebbian constraint at each learning step executed in this network. We find that the hidden internal model enables searches for cause-effect relationships in the form of autoregressive models under certain conditions. We discuss the connection between hidden causes and Independent Subspace Analysis. We speculate that conscious experience is the result of competition between various learned hidden models for spatio-temporal reconstruction of ongoing effects of the detected hidden causes.

Introduction

The homunculus fallacy, an enigmatic point of artificial general intelligence, has been formulated by many (see e.g., Searle 1992). It says that representation is meaningless without 'making sense of it', so the representation needs an interpreter. Then it continues with the questions: Where is this interpreter? What kind of representation is it using? This line of thoughts leads to an infinite regress. The problem is more than a philosophical issue. We are afraid that any model of declarative memory or a model of structures playing role in the formation of declarative memory could be questioned by the kind of arguments provided by the fallacy.

Our standpoint is that the paradox stems from vaguely described procedure of '*making sense'*. The fallacy arises by saying that the internal representation should make sense. To the best of our knowledge, this formulation of the fallacy has not been questioned except in our previous works (see, Lőrincz et al. (2002), and references therein). We distinguish input and the representation of the input. In our formulation, the '*input makes sense'*, if the representation can produce an (almost) identical copy of it. This is possible, if the network has experienced and properly encoded similar inputs into the representation previously. According to our approach, the internal representation interprets the input by (re)constructing it. This view is very similar to that of MacKay (1956) who emphasized *analysis and synthesis* in human thinking and to Horn's view (1977), who said that vision is inverse graphics.

In the next section, we build an architecture by starting from an auto-associative network that has input and *hidden representation*. We will insist on Hebbian learning for each transformation, i.e., from input to representation and from representation to *reconstructed input,* of the network. We will have to introduce additional algorithms for proper functioning and will end up with a network that searches for *cause-effect relationships.* During this exercise we remain within the domain of linear approximations. In the discussion we provide an outlook to different extensions of the network, including non-linear networks, and probabilistic sparse spiking networks. The paper ends with conclusions.

Making sense by reconstruction

We start from the assumption that the representation 'makes sense' of the input by producing a similar input. Thus, steps of making sense are:

1. input → representation
2. representation → reconstructed input

If there is a good agreement between the input and the reconstructed input then the representation is appropriate and the input 'makes sense'. Observe that in this construct there is no place for another interpreter, unless it also has access to the input. However, there is place for a hierarchy, because the representation can serve as the input of other reconstruction networks that may integrate information from different sources. A linear reconstruction network is shown in Fig. 1. We note that if the model recalls a representation, then it can produce a reconstructed input in the absence of any real input.

First, we shall deal with static inputs. Then we consider inputs that may change in time.

The Case of Static Inputs

We start from the constraints on the representation to reconstructed input transformation. The case depicted in Fig. 1 corresponds to Points 1 and 2 as described above. However, it requires a slight modification, because we

need to compare the input and the reconstructed input. This modification is shown in Fig. 2.

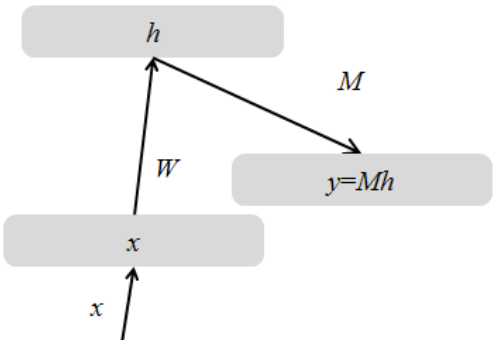

Figure 1. $x \in R^n$: We say that input layer has *n* neurons. The activity of the i^{th} neuron is x_i. $h \in R^n$: there are *n* neurons in the hidden representation layer. $W \in R^{n \times n}$: input-to-representation, or bottom-up (BU) transformation. W_{ij} is the ij^{th} element of matrix W: 'synapse' or weight from neuron *j* to neuron *i*. $y \in R^n$: there are *n* neurons in the reconstructed input layer. $M \in R^{n \times n}$: top-down (TD) transformation.

Input $x \in R^n$ is compared with the reconstructed input $y \in R^n$ and produces the reconstruction error $e \in R^n$. Then, reconstruction error can be used to correct the representation. It is processed by bottom-up (BU) matrix $W \in R^{n \times n}$ and updates the representation $h \in R^n$. Representation is processed by top-down (TD) matrix $M \in R^{n \times n}$to produce the reconstructed input. The relaxation dynamics is:

$$h(t + \Delta t) = I\, h(t) + W(x(t) - Mh(t)) \quad (1)$$

$$h(t) \approx \int_{-\infty}^{t} \exp(-WM(t - \tau))Wx(\tau)d\tau \quad (2)$$

Note that update (1) requires a recurrent synapse system that represents the identity matrix I to add $h(t)$ to the update $W(x(t) - Mh(t))$ at time $t + 1$. We will come back to this point later.

Equation (2) is stable if $WM > 0$ (WM is positive definite). Then the architecture solves equation $x = Mh$ for *h*, so it effectively computes the (pseudo-)inverse, provided that the input is steady. Even for steady input, condition $WM > 0$ should be fulfilled, so we have to train matrix M. Training aims to reduce the reconstruction error and we get cost function $J(M) = \frac{1}{2}\sum_t |x(t) - Mh(t)|^2$ and then the on-line tuning rule:

$$\Delta M \propto \varepsilon(t)h(t)' \quad (3)$$

where apostrophe denotes transpose and $\varepsilon(t) = x(t) - y(t)\ \big(= x(t) - Mh(t)\big)$.

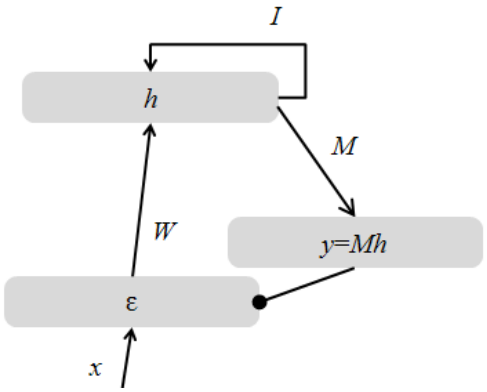

Figure 2. $\varepsilon \in R^n$: the input layer receives inhibitory (negative) feedback from the reconstructed input and becomes a comparator. The input layer holds the reconstruction error ε. Arrow with solid circle: additive inhibition.

We have to modify Fig. 2 to make this learning rule Hebbian (Fig. 3):

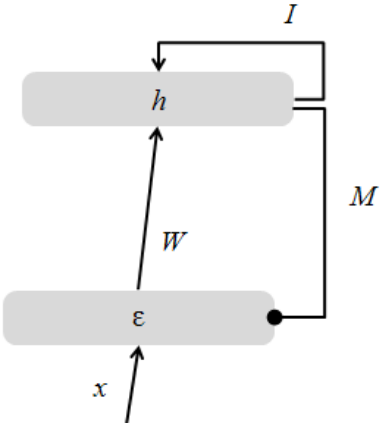

Figure 3. Hebbian training for TD matrix M (Eq. (3)).

Clearly, training of matrix M stops if $M = W^{-1}$, which includes the trivial solution, $M = W = I$. Condition $WQ > 0$ is satisfied. The situation is somewhat more delicate if input may change by time. We treat this case below.

The Case of Inputs that Change by Time

If inputs change by time, then we can not reconstruct them, because of two reasons (i) there are delays in the reconstruction loop and (ii) the network may need considerable relaxation time if matrix Q is not properly tuned. We have to include predictive approximations to overcome these obstacles.

First, we introduce a predictive model. Second, we discover problems with Hebbian learning that we overcome by means of the representation. New Hebbian problems will constrain us that we solve by another rearrangement of the network.

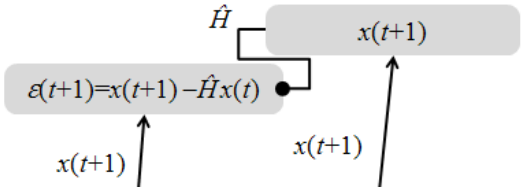

Figure 4. Hebbian learning for predictive matrix $\hat{H}$.

For the sake of simplicity, we assume that the input is a first order autoregressive model (AR(1)):

$$x(t+1) = Hx(t) + n(t) \tag{4}$$

where $H \in R^{n\times n}$ and its largest eigenvalue is smaller than 1 (for stability) and $n \in R^n$ is the driving noise having normal distribution. Our approximations are $\hat{H}$ for matrix *H*, and $\hat{x}$ for input estimation, i.e., we estimate $x(t+1)$ as

$$\hat{x}(t+1) = \hat{H}x(t) \tag{5}$$

and the estimation error is

$$\varepsilon(t+1) = x(t+1) - \hat{x}(t+1) \tag{6}$$

and ε is our estimation for noise *n*. Our cost function is $J((H)) = \frac{1}{2}|x(t+1) - \hat{H}x(t+1)|^2$ that leads to the Hebbian training rule:

$$\Delta\hat{H} \propto \varepsilon(t+1)x(t)' \tag{7}$$

The network that can realize Eq. (7) is shown in Fig. 4.

The network in Fig. 4 works as follows. Input $x(t)$ arrives to the two input layers and starts to propagate through matrix $\hat{H}$. At the next time instant input $x(t+1)$ arrives and the propagated input is subtracted, so we have activities $\varepsilon(t+1) = x(t+1) - \hat{H}x(t)$ on the output end of matrix $\hat{H}$ and the synapses were traversed by $x(t)$, satisfying the constraints of rule (7).

There is a *problem* with the network of Fig. 4: we can not ensure identical inputs at different layers. This problem can be solved if we insert this new network into our previous two-layer architecture (Fig. 3). Having done this, for time varying inputs Eq. (3) assumes the form

$$\Delta M \propto \varepsilon(t+1)h(t)' \tag{8}$$

As we shall see, Eq. (8) enables the learning of a hidden model.

Two layer network with hidden predictive matrix. We add a predictive model (matrix $F \in R^{n\times n}$) to the representation layer; it replaces the identity matrix I as required by non-steady inputs (Fig. 5). Now, we examine how this matrix could be trained.

Equation $M = W^{-1}$ still holds, provided that matrix F – our estimated model – can compensate for the temporal changes. The model at the representation layer is:

$$h(t+1) = Fh(t) + n_h(t+1), \tag{9}$$

where according to our notations, noise n_h should be an estimation of Wn.

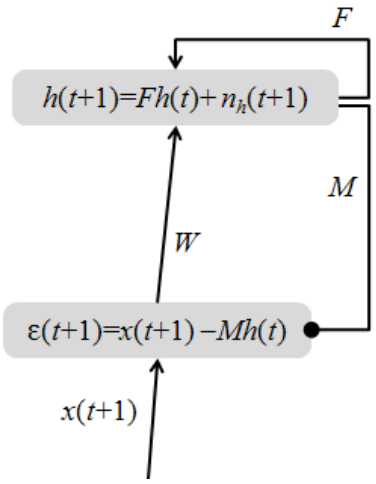

Figure 5: Representation with predictive model.

The question we have is whether we can learn a non-inhibitory prdictive matrix F by Hebbian means or not. Although we can learn predictive matrices, see, e.g., Eq. (7), but they would work as comparators.

For model learning, the same trick does not work, we need other means. Our simple structure can be saved if we assume two-phase operation. It is important that two-phase operation fits neuronal networks (Buzsáki, 1989), so we are allowed to use this trick. We assume that $x(t)$ and $\varepsilon(t+1)$ are transferred in Phase I and Phase II respectively by bottom-up matrix W. Under this condition, training of predictive matrix F can be accomplished in Phase II: in

Phase II, the output of matrix is $FWx(t)$, whereas it experiences input $W\varepsilon(t+1)$. The same quantities emerge when considering cost $J(F) = \frac{1}{2}|h(t+1) - Fh(t)|^2$, i.e., the squared error $n_h(t)$ at time t. Note, however, that training of matrix F is supervised and so matrix F can play an additive role.

Discussion

The resolution of the homunculus fallacy has been suggested in our previous works (see, e.g., Lőrincz et al. (2002), and references therein). Here we elaborated that work by more rigorous considerations on Hebbian learning. We were led to a simple network that provides further insights into the 'making sense' process:

(1) The network discovers two components: (i) a deterministic process characterized by the predictive matrix and (ii) the driving noise of this deterministic process. One may say that the network discovers the causes (the driving noises) and the effects (the deterministic evolution of the driving noises).

(2) The network builds up an internal model that can run without input. Assume that the network runs for k steps on its own

$$\hat{h}(t+k) = F^k h(t) \tag{10}$$

and then it compares the result with the input k steps later:

$$\varepsilon_k(t+k) = x(t+k) - M\hat{h}(t+k) \tag{11}$$

If the disagreement between the two quantities is small (if $\varepsilon_k(t+k)$ that appears at the input layer is small), then the input process 'makes sense' according to what has been learned.

We note for the sake of arguments on consciousness that if the network runs for *k* time steps, then – according to the dimensional constraints – the network can be increased up to *k* pieces of parallel running temporal processes, each of them trying to reconstruct the input during the whole *k* time step history. The pseudo-inverse method is suitable to select the sub-network with the smallest reconstruction error over the *k* time steps. This sub-network makes the most sense according to history.

(3) The same predictive network can be used for replaying temporal sequences, provided that the starting hidden representation is saved somewhere.

The novelty of this work comes from the examination of Hebbian constraints on reconstruction networks. Neural networks with reconstruction capabilities, however, are not new; there is long history of such networks.

Other works starting from similar thoughts

There are many network models that have similar structure. These networks are typically more complex than the simple/minimal linear autoregressive network that we described here. There are similar networks that aim to model real neuronal architectures. The literature is huge; we can list only some of the most prominent works.

To our best knowledge, the first neocortex related reconstruction network model that suggested approximate pseudo-inverse computation for information processing *between* neocortical areas was published by Kawato et al., (1993). It was called the *forward-inverse model* and modeled the reciprocal connections between visual neocortical areas. The motivation of the model was to connect regularization theories of computational vision (Poggio et al., 1985, Ballard et al., 1983) to neocortical structure and explain how multiple visual cortical areas are integrated to allow coherent scene perception. The computational model of the neocortex was extended by Rao and Ballard (Rao and Ballard, 1997, Rao and Ballard, 1999), who suggested that neocortical sensory processing occurs in a *hierarchy of Kalman filters*. The Kalman filter model extends previous works into the temporal domain.

Non-linear extensions include the so called recurrent neural networks that have non-linear recurrent collaterals at the representation layer. For a review on recurrent neural networks, see Jacobsson (2005). A particular recurrent network model with hidden layer is called Echo State Network (ESN, Jaeger, 2003). ESN – unlike to most models – is non-linear with strictly Hebbian learning. It does not assume two-phase operation. It is made efficient by a huge random recurrent network that forms the internal representation.

Another type of networks with reconstruction flavor belongs to stochastic networks and is called generative model (see, e.g., (Hinton, 2007). An attempt that connects generative models with two phase operation appeared early (Hinton, 1995), but without details on Hebbian constraints.

The Kalman filter model and the generative network model are the close relatives of the minimal architecture that we described here. They are more sophisticated, but Hebbian learning is so strict as in our minimal model.

Extensions of reconstruction networks

The role of the bottom-up matrix. It is intriguing that Hebbian learning did not provide constraints for the bottom-up matrix. Our proposal, that hidden models discover cause-effect relationships (see point (1) above), leads to the thought that the role of the bottom-up matrix is to help searches for causes. Causes – by definition – are independent, so we have to look for independent sources.

This route is relevant if the noise is not normal, which the typical case for natural sources is. If non-normal sources are hidden and only their mixture is observed, then observed distribution may approximate a normal distribution, because of the d-central limit theorem. Then the following situation is achieved:

1. Deterministic prediction can be subtracted from the observation under the assumption that the driving noise is close to normal distribution
2. Independent sources can be estimated by independent subspace analysis (see, e.g., Cardoso (1998), Hyvarinen and Hoyer (2000)). For a review, see Szabó et al. (2007).
3. The autoregressive processes in the independent subspaces can be learnt by supervisory training that overcomes the problem of non-normal distributions. We note: (a) the least mean square approach that we applied fits the normal distribution, (b) higher order autoregressive processes with moving averages can also be included into the representation (Szabó et. al, 2007, Póczos et. al, 2007), although it is not yet known how to admit Hebbian constraints.
4. It is unclear if Independent Subspace Analysis can be performed by Hebbian means or not. Efforts to find strictly Hebbian methods for the whole loop including the independent subspace analysis are in progress (Lőrincz et al., 2008a).

The search for cause-effect dependencies can be related to the *Infomax* concept (Barlow, 1961, Linsker, 1988, Atick and Redlich, 1992, Bell and Sejnowski, 1995, Linsker, 1997), because upon removing the temporal process, the search for the *independent* causes is analogous to the Infomax concept (Cardoso, 1997). However, the reasoning is different; here, the aim of independent subspace analysis is to find the causes that drive deterministic processes.

Extensions of this simple architecture to ARMA(p,q) processes (Póczos et al., 2007), non-linear extensions (Jaeger, 2003), extensions with control and reinforcement learning (Szita and Lőrincz, 2004, Szita et al., 2006) are possible. Overcomplete probabilistic sparse spiking extension of the reconstruction architecture has also been suggested (Lőrincz et al., 2008b) and this direction has promises for biologically plausible probabilistic spatio-temporal extensions of the 'making sense procedure' under Hebbian constraints.

Outlook to a potential model for consciousness. It has been mentioned before that if the model runs without input for k steps, then the number of models can be multiplied by k, because the pseudo-inverse method can select the best candidate. There is a cost to pay: the best process can not be switched off arbitrarily often, it should be the best candidate that reconstructs k time steps. Such competition between models to represent the sensory information may explain certain aspects of consciousness, including rivalry situations, when perception is changing steadily, whereas the sensory information is steady.

Conclusions

We have shown that under Hebbian constraints, the resolution of the homunculus fallacy leads to a particular reconstruction network. The network is potentially the simplest in its structure, but not in its functioning: (i) it has a bottom-up, a top-down, and a predictive network, and it is linear, but (ii) it works in two separate phases.

We have shown that the emerging network turns the philosophical infinite regress into a finite loop structure and this finite loop uncovers hidden cause-effect relationships. This is one way to interpret the making sense procedure of the 'homunculus'. The representation produces the next expected input from time-to-time and computes the difference between the input and this expected reconstructed input. We say that the input makes sense, if this difference is within the range of the expected noise. Also, the network can run by itself as required if inputs are missing.

We have found that constraints arising from the resolution of the fallacy leave the form of the bottom-up network open. However, the reconstruction network uncovers hidden deterministic processes and estimates the driving noise, the hidden causes. Causes are independent 'by definition', so the network should work better if the bottom-up transformation is trained on the estimated noise according to Independent Subspace Analysis (ISA), which is provably non-combinatorial under certain circumstances (Póczos et al., 2007, Szabó et al., 2007). The concept of finding causes that drive deterministic processes leads to and takes advantage of a relative of ISA, the so called Infomax concept, which has been developed for modeling sensory information processing in the brain (Barlow 1961, Linsker 1988).

We have speculated that competing models in reconstruction networks may provide a simple explanation for certain features of consciousness. This speculation can be taken further: the model hints that conscious experience may emerge as the result of *distributed* and *self-orchestrated competition* amongst predictive models to reconstruct their common inputs over longer time intervals. This line of thoughts suggests to seek not (only) the *conductor of the orchestra* (see, e.g., Crick and Koch, 2005), but the *distributed selection algorithm* triggered by *unexpected independent causes* as disclosed by reconstruction errors of *competing reconstruction models*.

References

Atick, J. J. and Redlich, A. N. 1992. What does the retina know about natural scenes? *Neural Comput.* 4:196-210.

Ballard, D. H., Hinton, G. E., and Sejnowski, T. J. 1983. Parallel visual computation. *Nature*, 306:21-26.

Barlow, H. 1961. Possible principles underlying the transformations of sensory messages. In: *Sensory Communication*, W. Rosenblith (ed.), pp. 217-234. MIT Press, Cambridge, MA.

Bell, A. J. and Sejnowski, T. J. 1995. An information-maximization approach to blind separation and blind deconvolution. *Neural Comput.* 7:1129-1159

Buzsáki, Gy. 1989. A two-stage model of memory trace formation: a role for "noisy" brain states. *Neuroscience* 31: 551–570.

Cardoso, J.-F. 1997. Infomax and maximum likelihood for source separation, *IEEE Letters on Signal Processing*, 4: 112-114.

Cardoso, J.-F. 1998. Multidimensional independent component analysis. In *Proc. of Int. Conf. on Acoustics, Speech, and Sign. Proc.* Seattle, WA, USA. 4: 1941–1944.

Crick, F. C. and Koch, C. 2005. What is the function of the claustrum? *Phil. Trans. R. Soc. B* 360: 1271-1279.

Hinton, G., E. 2007. To recognize shapes, first learn to generate images. *Prog. Brain. Res.* 165:535-547.

Hinton, G. E., Dayan, P., Frey, B. J., and Neal, R. 1995. The wake-sleep algorithm for self-organizing neural networks. *Science*, 268:1158-1161.

Horn, B. 1977. Understanding image intensities. *Artificial Intelligence* 8: 201–231.

Hyvarinen, A., Hoyer, P.O. 2000. Emergence of phase and shift invariant features by decomposition of natural images into independent feature subspaces. *Neural Comput.*, 12: 1705–1720.

Jacobsson, H., 2005. Rule Extraction from Recurrent Neural Networks: A Taxonomy and Review. *Neural Comput.,* 17:1223-1263.

Jaeger, H. 2003, Adaptive nonlinear system identification with echo state networks, *Adv. in Neural Information Proc. Systems* 15: 593-600.

Kawato, M., Hayakawa, H., and Inui, T. 1993. A forward-inverse model of reciprocal connections between visual neocortical areas. *Network*, 4:415-422.

Linsker, R. 1988. Self-organization in a perceptual network. *IEEE Computer* 21:105-117.

Linsker, R. 1997. A local learning rule that enables information maximization for arbitrary input distributions. *Neural Comput.* 9:1661-1665.

Lőrincz, A. Kiszlinger, M., Szirtes, G. 2008a. Model of the hippocampal formation explains the coexistence of grid cells and place cells. http://arxiv.org/pdf/0804.3176

Lőrincz, A., Palotai, Zs., Szirtes, G., 2008b. Spike-based cross-entropy method for reconstruction. *Neurocomputing*, 71: 3635-3639.

Lőrincz, A., Szatmáry, B., and Szirtes, G. 2002. Mystery of structure and function of sensory processing areas of the neocortex: A resolution. *J. Comp. Neurosci.* 13:187–205.

MacKay, D., 1956. Towards an information-flow model of human behavior. *British J. of Psychology* 47: 30–43.

Póczos, B., Szabó, Z., Kiszlinger, M., Lőrincz, A. 2007. Independent Process Analysis Without a Priori Dimensional Information. *Lecture Notes in Comp. Sci.* 4666: 252–259.

Poggio, T., Torre, V., and Koch, C. 1985. Computational vision and regularization theory. *Nature*, 317:314-319.

Rao, R. P. N. and Ballard, D. H. 1997. Dynamic model of visual recognition predicts neural response properties in the visual cortex. *Neural Comput.*, 9:721-763.

Rao, R. P. N. and Ballard, D. H. 1999. Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects. *Nature Neurosci.*, 2:79-87.

Searle, J., 1992 *The Rediscovery of Mind.* Cambridge, MA.: Bradford Books, MIT Press.

Szabó, Z., Póczos, B., Lőrincz, A. 2007. Undercomplete Blind Subspace Deconvolution. *J. of Machine Learning Research* 8: 1063-1095.

Szita, I., Gyenes, V., Lőrincz, A. 2006. Reinforcement Learning with Echo State Networks, ICANN 2006, *Lect. Notes in Comp. Sci.* 4131: 830–839.

Szita, I., Lőrincz, A., 2004. Kalman filter control embedded into the reinforcement learning framework. *Neural Comput.* 16: 491-499.