RESEARCH ARTICLE

Dynamic searching in the brain

Eduardo Mizraji · Andrés Pomi · Juan C. Valle-Lisboa

Abstract Cognitive functions rely on the extensive use of information stored in the brain, and the searching for the relevant information for solving some problem is a very complex task. Human cognition largely uses biological search engines, and we assume that to study cognitive function we need to understand the way these brain search engines work. The approach we favor is to study multimodular network models, able to solve particular problems that involve searching for information. The building blocks of these multimodular networks are the context dependent memory models we have been using for almost 20 years. These models work by associating an output to the Kronecker product of an input and a context. Input, context and output are vectors that represent cognitive variables. Our models constitute a natural extension of the traditional linear associator. We show that coding the information in vectors that are processed through association matrices, allows for a direct contact between these memory models and some procedures that are now classical in the Information Retrieval field. One essential feature of context-dependent models is that they are based on the thematic packing of information, whereby each context points to a particular set of related concepts. The thematic packing can be extended to multimodular networks involving input-output contexts, in order to accomplish more complex tasks. Contexts act as passwords that elicit the appropriate memory to deal with a query. We also show toy versions of several 'neuromimetic' devices that solve cognitive tasks as diverse as decision making or word sense disambiguation. The functioning of these multimodular networks can be described as dynamical systems at the level of cognitive variables.

Introduction

The human brain is the residence of distributed networks of neurons, with a large-scale regular connectivity. This large-scale neuroanatomical connectivity is the result of complex morphogenetic processes that occur during the embryonic development, and its regularity is evinced in that it can be described in standard textbooks (see, for instance, Delmas and Delmas 1958). Textbook descriptions are an empirical proof of the existence of this large-scale anatomical regularity. But these physical, regular networks sustain other networks: cognitive networks that connect data. Cognitive nets differ from one individual to another, and are based on memory banks which are installed through fine-scale variations in neural connectivity displayed by certain "trainable" synapses (Kandel and Schwartz 1985). These trainable synapses have signal transduction abilities modulated by somatic and sensory experiences. Hence, the memory modules are neuronal networks with trainable synapses open to be programmed by sensory inputs or inputs coming from other modules. In contrast with the large-scale regularity of neuroanatomical networks, the cognitive networks (e.g., the semantic- or conceptual-networks), although restricted by the underlying biology, are highly idiosyncratic. Much of the regularity displayed by the semantic networks of members of a community is due to the shared social and linguistic environments they are exposed to (Spitzer 1999, Chapter 10).

E. Mizraji (✉) · A. Pomi · J. C. Valle-Lisboa
Group of Cognitive Systems Modeling, Biophysical Section,
Facultad de Ciencias, Universidad de la República, Iguá 4225,
Montevideo 11400, Uruguay
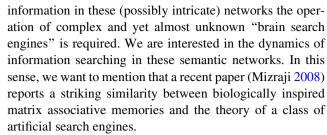e-mail: mizraj@fcien.edu.uy

Human brains are also information-processing physical devices, and their dynamical behavior displays a subtle connection with the meaning of the processed information. This meaning (or semantics) depends on biological conditions imposed at the same time by the cultural and evolutionary-developmental histories of individuals. The evolutionary and the developmental histories roughly map on the neural "hardware" represented by neuroanatomical networks. Inside these neuroanatomical networks the generation of propagated electrochemical signals between neurons occurs, and these signals are largely responsible for the complex neurocomputational events that the brain can perform (for a comprehensive analysis, see Wright et al. 2004 and also Tsuda 2001). Cultural heritage maps on the programmed brain, a kind of "software" installed by learning.

There is thus a form of "dynamical complementarity". On the one hand, cognitive neural systems are physical machines with highly complex dynamics that in many cases can be described using systems of differential or difference equations. These descriptions are based on time-dependent physicochemical variables. On the other hand, the system can be described as supporting a dynamic behavior governed by semantics. In this second framework, the dynamical system involves variables representing meanings. Obviously both complementary levels must be physically consistent, and in the neural domain these levels do not lead to conflicting descriptions (beim Graben et al. 2008b). Indeed, the biophysical neuronal dynamics that occur during cognitive processes triggered by neural queries coexist with the dynamics of semantic neural vectors governed by the structure of the neural databases and by the semantic contexts involved in the neural queries.

The understanding of what we call dynamic complementarity is a technically difficult matter that requires intensive explorations. A preliminary approach to model this complementarity was attempted by Mizraji and Lin (1997, 2001) for the case of logical decisions. In these works, the simplified representation of neural activities involves large dimensional vectors that represent logical decisions; these vectors interact with associative memories storing logical gates. Semantics is implied by the structure of the memories which is established by the developmental history of the individual (in a large sense, this development includes learning). For another view of this complementarity see Pomi and Mizraji (2004).

Language is a privileged object to analyze this type of problems (Elman 1990, 1993) and, very recently, aspects of language as a dynamical cognitive system have been modeled by beim Graben et al. (2008a, b). Most of the processes that can be described at the level of cognitive dynamics, like language processing and production or even thinking processes are based on semantic networks. To find the relevant information in these (possibly intricate) networks the operation of complex and yet almost unknown "brain search engines" is required. We are interested in the dynamics of information searching in these semantic networks. In this sense, we want to mention that a recent paper (Mizraji 2008) reports a striking similarity between biologically inspired matrix associative memories and the theory of a class of artificial search engines.

In the present work, we propose a framework to describe and analyze biological "dynamic search engines" based on matrix memories that create thematic blocks using multiplicative contexts. These memories are supported by modular networks and are accessed using passwords represented by vectorial semantic contexts. A cognitive query produces an output that is distributed between many memory modules due to the anatomic connectivity. A given query can trigger different associative trajectories according to the sequence of contexts involved.

In order to show the versatility of this approach, we describe modular architectures whose modules are neural networks capable of supporting a variety of functions, including associative memories structured on semantics. In this framework, the neural dynamics is governed by the association between inputs $X(t)$ and outputs $Y(t + \tau) = F(X(t), P)$, where X and Y represent vectorial variables that code for the relevant neural activities and P represents a vectorial control parameter. The function F is usually implemented through an associative memory. In most cases we are going to assume that a fixed time interval τ elapses between the input and the output of a memory operator. Consequently, in the cases we are going to consider in this work, we usually omit explicit representations of time t. After showing the basic workings and properties of our kernel model, we describe some interesting extensions to output context modulation at the end of Section Distributed memories. In Section Memory modules and latent semantics we establish a connection between matrix memory models and information retrieval procedures. After that, in Section Multimodular memory systems we show that an intersection filter based on a context dependent module can be useful in a variety of tasks, ranging from diagnosis on the face of partial information (as illustrated with medical diagnosis) to adaptive meaning disambiguation.

Distributed memories

The operation of neural memories can be modeled using vector spaces and matrix algebra (Kohonen 1972, 1977; Anderson 1972; Cooper 1973, 2000). In this framework, a modular neural memory can be represented as an operator linking two vector spaces: an m-dimensional input space and an n-dimensional output space.

Vector codes

The elements of the vector patterns are real numbers corresponding to the electrochemical signals transported by the neuronal axons (usually, frequencies of action potentials). For instance, an optical pattern is transduced into action potentials, and transferred inside the brain through thousands of axons displaying its own frequency of action potentials. This large set of different frequencies is the immediate neural codification. This original vector is further processed in other neural centers and information ends up coded by other neural vectors. Consequently, a memory that associates faces with names, receives as input a vector that results from many processing steps. The pronunciation or writing of the name triggered by the face depends on a neural vector that (after processing) activates the muscular effectors responsible for speech or writing.

Let $\hat{f} \in R^m$ and $\hat{g} \in R^n$ be the input and the output column vectors processed by an associative memory Mem. The installation of a database of K associated patterns $\hat{f}_i$, $\hat{g}_i$ in a memory implies that

$$\hat{g}_i = \text{Mem}(\hat{f}_i) \ (i = 1, 2, \ldots, K). \tag{1}$$

The operators like Mem are functional relations generated by the learning processes, and they are the key functions governing the cognitive dynamics sustained by neuronal networks.

Matrix memory models

In the simplest case the memory operator Mem can be represented by the matrix

$$M = \sum_{i=1}^{K} \hat{g}_i \hat{f}_i^{T} \tag{2}$$

(Kohonen 1977). This matrix format assumes non-linear neurons that produce, in the frequency domain, an input-output relation approximately described by

$$r(\alpha) = \Phi \left[\sum_{\beta=1}^{m} M_{\alpha\beta} s(\beta) - U(\alpha) \right] \sum_{\beta=1}^{m} M_{\alpha\beta} s(\beta), \tag{3}$$

where $M_{\alpha\beta}$ is the weight of the synapse connecting axon β with neuron α, $U(\alpha)$ is the threshold of neuron α, $s(\beta)$ is the frequency of action potentials coming via axon β, and $r(\alpha)$ is the integrated output of neuron α in the following time step. $\Phi[x]$ is the Heaviside function (for an in-depth analysis of this kind of models, see Nass and Cooper 1975). In the matrix model any network neuron is assumed to be physiologically submerged in noise displaying basal activities $r_0(\alpha)$ and $s_0(\beta)$, such that neurons are placed in the linear region. The final variables used to state the matrix expressions are $g(\alpha) = r(\alpha) - r_0(\alpha)$ and $f(\beta) =$

$s(\beta) - s_0(\beta)$ that are the scalar components of vectors $\hat{g}$ and $\hat{f}$.

The cardinal property of these matrix memories is the scalar product filtering:

$$M\hat{f} = \sum_{i=1}^{K} \langle \hat{f}_i, \hat{f} \rangle \hat{g}_i, \tag{4}$$

$\langle a, b \rangle = a^T b$ (scalar product between column vectors a and b).

We illustrate now how the scalar product determines the pattern recognition. If the set $\{\hat{f}_i\}$ is orthogonal, and the input $\hat{f} = \hat{f}_K \in \{\hat{f}_i\}$, then recollection of the memory M is exact (except for a scale factor):

$$M\hat{f}_k = v_k \hat{g}_k, \tag{5}$$

$v_k = \|\hat{f}_k\|^2$ ($\|z\| = \sqrt{\langle z, z \rangle}$ is the Euclidean norm of vector z).

In general, if $\hat{f} \in S_M$ the output is a linear combination of the $\hat{g}_i$; if $\hat{f}$ is orthogonal to S_M (we put: $\hat{f} \perp S_M$) there is no recognition: $M\hat{f} = 0$.

We are going now to express the matrix memories using normal vectors. For $f_i = \hat{f}_i \cdot \|\hat{f}_i\|^{-1}$ and $g_i = \hat{g}_i \cdot \|\hat{g}_i\|^{-1}$ the matrix M becomes

$$M = \sum_{i=1}^{K} \mu_i g_i f_i^{T}, \tag{6}$$

with

$$\mu_i = \|\hat{g}_i\| \cdot \|\hat{f}_i\|.$$

Let In = $\{f_i\}$ and Out = $\{g_i\}$ be the Input and the Output sets. If In and Out are orthonormal sets, that matrix M satisfies the singlular value decomposition (SVD) conditions:

(a) $Mf_k = \mu_k g_k$

(b) $M^T g_k = \mu_k f_k$

for $f_k \in$ In and $g_k \in$ Out. Consequently, if these conditions are satisfied, then Eq. 6 represents the SVD of M, being the μ_k the singular values, and f_k and g_k their associated singular vectors. In realistic cases, the output or input vectors need not be orthogonal and the matrix structure in terms of associated normal vectors can only be an approximation to the true SVD.

Multiplicative contexts

In real biological memories, semantic contexts condition the associations triggered by a stimulus. The sensitivity to semantic contexts is a way to disambiguate information susceptible of many different interpretations (e.g., polysemic words). To obtain a mathematical representation of a context-sensitive memory it is necessary to find an operation able to associate a pair of input vectors (f, p) with a vector output g, i.e. a function of the form $g = T(f, p)$. An

important early work, describing a double filtering procedure, was published by Pike (1984). In the late 1980s different authors described similar procedures to solve this problem (see Mizraji et al. 1994 and Valle Lisboa et al. 2005 for a statement of the problem and the review of other authors' contributions).

Consider a memory E given by

$$E = \sum_i g_i (p_i \otimes f_i)^T \tag{7}$$

that stores superimposed trace memories i that associate an output g_i with a pair of inputs, f_i and p_i, acting as a key stimulus and their context. The symbol $\otimes$ represents the Kronecker product (Graham 1981). The Kronecker product could result from the synapses of these neurons working as coincidence detectors (Mizraji 1989; Pomi and Mizraji 1999; Fig. 1).

The key property of this multiplicative contextualization is the double filtering, where an input of the form $p \otimes f$ is processed as:

$$E(p \otimes f) = \sum_i \langle p_i, p \rangle \langle f_i, f \rangle g_i, \tag{8}$$

These multiplicative contexts open many computing capabilities (Mizraji 1989; Pomi and Mizraji 1999, 2004) derived from their ability to perform adaptive associations. It is also remarkable that, for large dimensionalities as those expected from neurobiological patterns of activity, these computing abilities remain operational in the case of incomplete Kronecker products (with many vector components missing). This is an important fact from a biological point of view (Pomi and Mizraji 1999) since it relaxes the stringent requirements that the full Kronecker product imposes.

(a) Virtual memories

It is interesting to consider that a context-dependent memory is the result of the superposition of many associative
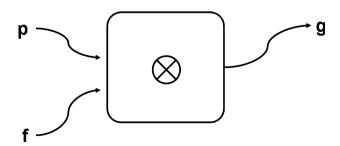

Fig. 1 Elementary multiplicative module. This module represents a complex associative memory capable of performing a Kronecker product between two input vectors (or a "statistical" Kronecker product where some elements are absent)

memories. This fact is clearly illustrated in the following argument. Let

$$E = \sum_{i,j} \mu_{ij} g_{ij} (p_i \otimes f_i)^T \tag{9}$$

be a multiplicative matrix memory where the weighted output g_{ij} is associated with the input f_j and the context p_i. The Kronecker product allows us to express an input as $p_k \otimes f = (p_k \otimes I)f$.

Consequently,

$$E(p_k \otimes f) = M^{(k)} f, \tag{10}$$

where $M^{(k)}$ is a "virtual memory" (Pomi and Mizraji 1999) with the structure

$$M^{(k)} = \sum_{i,j} \mu_{ij}(k) g_{ij} f_j^T, \tag{11}$$

with $\mu_{ij}(k) = \mu_{ij} \langle p_i, p_k \rangle$.

Notice that if $\{f_j\}$ and $\{g_{ij}\}$ are orthonormal sets these $\mu_{ij}(k)$ are the singular values of $M^{(k)}$, and the action of the contexts could be seen as the instruction of different spectra corresponding to a same set of associations stored in a matrix memory $M^{(k)}$.

(b) Intersection filters

An interesting property of these networks emerges from the multiplicative version of an autoassociative matrix memory. In the presence of a working memory capable of retaining previous data, the memory we are going to considered is able to act as an intersection filter. Given the matrix memory

$$Y = \sum_{i=1}^{R} a_i (a_i \otimes a_i)^T, \tag{12}$$

let an actual input $a_1 + a_6 + a_8$ be contextualized by a vector retained in the working memory having the structure $a_2 + a_6$. The output associated by this memory is

$$Y[(a_1 + a_6 + a_8) \otimes (a_2 + a_6)] = a_6.$$

This intersection filter has been used to solve, in terms of a modular neural network, a problem stated by M. Minsky referred to the refinement of diagnoses produced by a neural system from a flux of incoming partial cues (Pomi and Mizraji 2001), a situation that we describe in Section Multimodular memory systems.

Input and output contexts

We can extend the power of context-depending matrix memories if we assume that the outputs include multiplicative contexts. In this way, such an output is prepared to enter another context memory and to generate a context modulated associative trajectory.

(a) Context modulated associative trajectories

Hence, we define a memory $H = \sum_{i,j} \hat{d}_{ij}\hat{h}_{ij}^T$ with the input and the output structured as

Input : $\hat{h}_{ij} = \hat{p}_i \otimes \hat{f}_{ij}$,

Output : $\hat{d}_{ij} = \hat{p}'_i \otimes \hat{g}_{ij}$.

Note the following important property:

$$(\hat{p}'_i \otimes \hat{g}_{ij})(\hat{p}_i \otimes \hat{f}_{ij})^T = (\chi_i p'_i p_i^T) \otimes (\mu_{ij} g_{ij} f_{ij}^T)$$

being $\chi_i = \|\hat{p}'_i\| \cdot \|\hat{p}_i\|$ and $\mu_{ij} = \|\hat{g}_{ij}\| \cdot \|\hat{f}_{ij}\|$.

Consequently, the structure of an Input–Output Context-Dependent Memory is

$$H = \sum_i \left(\chi_i p'_i p_i^T \otimes \sum_j \mu_{ij} g_{ij} f_{ij}^T \right). \quad (13)$$

It is important to note that using the original vectors $\hat{h}_{ij}$ and $\hat{d}_{ij}$ the memory (13) can be written with the format of a simple memory (Eq. 2).

Defining $M_i = \sum_j \mu_{ij} g_{ij} f_{ij}^T$,
we have

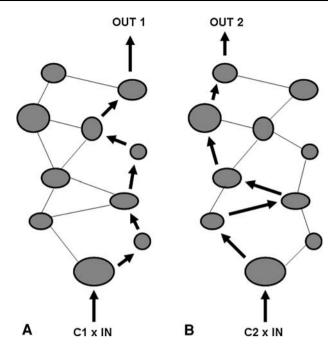$$H = \sum_i \left(\chi_i p'_i p_i^T \otimes M_i \right). \quad (14)$$

Note that these memory modules, as the other context-dependent matrix memories, are accessed by key inputs having a kind of password included in them, but at the same time, they generate outputs labeled with the corresponding context password. In Fig. 2 we illustrate how a given input, with two different contexts, can trigger two different associative trajectories inside a modular neural network.

We can imagine that devices of this kind can be useful to direct the outputs towards memory modules concerned with specific categories (themes). For instance, a polysemic word can be disambiguated because the subject of a conversation imposes input and output contexts that make the word to be refused by the thematic modules that are not relevant for the conversation, and to be accepted only by the correct associative module.

(b) Neuro-anatomical coding

The formalism using input and also output context vectors in associated pairs of memory traces enables, in the particular case we analyze hereafter, the representation of spatiality of memory modules and how the correct processing neural areas are selected according to the neural pathway by which the information arrives. In this sense, the input–output contexts formalism will also be a mathematical way of expressing the neuro-anatomical wiring.

If contexts p_i and p'_i are both the same n-dimensional unit vector e_i then

Fig. 2 Different searching trajectories in the same network. **a** and **b** represent the same modular network with a pre-established connectivity, being each module an input–output context associative memory. In **a** an input IN together with a context C1 produces the associative trajectory illustrated by the arrows, arriving to output OUT 1; instead, in **b**, the same input but now with context C2 produces a different associative trajectory generating an output OUT 2

$$e_i e_i^T = [\delta_{ki}\delta_{ij}] \in R^{n\times n} \quad i,j = 1, \ldots, n,$$

with $\delta_{\alpha\beta} = 1$ iff $\alpha = \beta$ and $\delta_{\alpha\beta} = 0$ iff $\alpha \neq \beta$. In this case, memory H is given by

$$H = \sum_{i=1}^s \left(e_i e_i^T \otimes M_i \right) = \begin{bmatrix} M_1 & 0 & \cdots & 0 \\ 0 & M_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & M_s \end{bmatrix}, \quad (15)$$

a diagonal matrix with different associative memories M_i, to be selected by a contextual environment e_i. In this way, this kind of context password carries positional information. The positions of the vector diag(H) addressed by the input-output context e_i, represent an arbitrary mapping of different spatial–anatomical–localizations of memories M_i, scattered through different sectors of cerebral cortex.

The interplay between input–output contexts e_i constitutes a representation of the neuroanatomical wiring that assures that the key stimulus addresses the corresponding memory module. Fed with entries of the type $e_k \otimes f_e$, the first term represents the anatomical pathway by which the information arrives: the neural wiring of information.

Notice that Eq. 15 admits a generalized version:

$$H = \sum_{i=1}^s \left(e_i e_i^T \otimes L_i \right),$$ where L represents a matrix memory

operator that could be, among others, a classical associative memory—it is the case of M shown in Eq. 15, or a context dependent memory matrix E as that presented in Eq. 9. In this last case, in each anatomical localization of the neural layer there is a set of superimposed associative memories able to be extracted by a local context p.

This representation captures the vision of the cortex as composed by anatomically different memory modules accessible by the adequate wiring, each one containing inside a set of superimposed distributed associative memories expressing themselves dynamically in the context of interactions with other areas (Friston 1998).

Memory modules and latent semantics

In this section we describe an important connection between neural memory models and the vector space models used to extract information from databases or corpora containing textual documents.

Contexts and thematic clusters

Consider now the following context-dependent memory E,

$$E = \sum_{i,j} \omega_{ij} d_{ij} \left(p_i \otimes \sum_{k=1}^{N_j} \sigma_{kj} f_{kj} \right)^T \tag{16}$$

where $\hat{p}_i \in R^{n'}$ is a context vector labeling a particular theme, $\hat{f}_{kj} \in R^n$ are the input patterns, $\hat{d}_{ij} \in R^m$ is the output associated to the patterns $\hat{f}_{kj}$ under the context $\hat{p}_i$; f_{kj}, P_i and d_{ij} are their normalized versions, with $\omega_{ij} = \left\| \hat{d}_{ij} \right\| \left\| \hat{p}_i \right\|$ and $\sigma_{ij} = \left\| \hat{f}_{kj} \right\|$.

We define an average input as follows:

$$\langle f_j \rangle = \frac{1}{N_j} \sum_{k=1}^{N_j} \sigma_{kj} f_{kj}. \tag{17}$$

Consequently, we can reduce the complexity of the expression defining
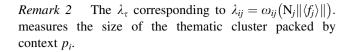
$$E = \sum_{i,j} \lambda_{ij} d_{ij} (p_i \otimes f_j)^T \tag{18}$$

with $f_j = \langle f_j \rangle \left\| \langle f_j \rangle \right\|^{-1}$ and $\lambda_{ij} = \omega_{ij} (N_j \left\| \langle f_j \rangle \right\|)$.

Finally we re-obtain the basic matrix structure:

$$E = \sum_{\tau=1}^{K} \lambda_\tau d_\tau h_\tau^T \tag{19}$$

with $\tau = \tau(i,j)$, $\tau = 1,2,\ldots,K$; $d_\tau = d_{ij}$; $h_\tau = p_i \otimes f_j$ and $\lambda_\tau = \lambda_{ij}$.

Remark 1   If the sets $\{d_\tau\}$ and $\{h_\tau\}$ are orthonormal, Eq. 19 is the SVD of matrix E.

Remark 2   The λ_τ corresponding to $\lambda_{ij} = \omega_{ij} (N_j \left\| \langle f_j \rangle \right\|)$. measures the size of the thematic cluster packed by context p_i.

Natural and artificial search engines

The matrix associative memories described by Eqs. 2, 13 and 16 represent situations with different intrinsic complexities, yet the three cases can adopt the format of Eq. 19. This mathematical fact establishes an interesting contact with the vector space model for information retrieval used in the modern theory of artificial search engines. In what follows, we are going to explore this contact.

One of the important approaches to the theory of artificial search engines is the vector space model (Berry and Browne 2005). This model begins defining a formal framework to code textual information. The main construct of this model is the "term-by-document matrix" (TD-matrix):

$$A = [d_1 d_2 \cdots d_q] \tag{20}$$

with $d_j \in R^p$ being document vectors and p the total number of words in the vocabulary. The elements of d_j measure the presence of each class of word.

The TD-matrix can be formatted as a memory. This is immediately apparent from the SVD of A:

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T \quad \text{or} \quad A^T = \sum_{i=1}^{r} \sigma_i v_i u_i^T \ (r = \text{rank}(A));$$

$u_i \in R^p$ and $v_i \in R^q$.

It is clear that these matrices display the structure of a memory associating vectors u_i and v_i. The associated vectors condense information concerning sets of related documents including information often regarded as noise. It is desirable in these procedures to eliminate the noise, something that is achieved through dimensionality reductions. We comment two approaches that produce interesting dimensionality reductions:

(a) *Classical Latent Semantic Analysis* (Deerwester et al. 1990; Berry et al. 1995). This is one of the most important procedures, and the basic idea is to truncate the SVD retaining only the first k largest singular values:

$$A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T; k \ll r. \tag{21}$$

The theoretical explanation of the capabilities of this procedure has only recently been advanced (Papadimitriou et al. 2000; Ando and Lee 2001).

(b) *Selection of non-adjacent singular values* (Valle-Lisboa and Mizraji 2007). This procedure reduces the

dimensions of the problem selecting a set of leading singular vectors that can be considered as thematic labels. The resulting matrix is

$$A_L = \sum_{i=1}^{s} \sigma_{k(i)} u_{k(i)} v_{k(i)}^T,\qquad(22)$$

where $1 \leq \kappa(i) \leq r$, $\kappa(i)$ being the position of the thematic label, $i = 1, \ldots, s$. The value of s measures the total amount of subjects, normally being $s \ll r$. This procedure is based on the Perron–Frobenius Theory.

What is a query within this format? We define a query as a vector $d' \in R^p$ that represents a document containing a small group of keywords. Asking for documents with a query d' means to operate as follows,

$$A^T d' = \sum_{i=1}^{r} \sigma_i v_i \langle u_i, d' \rangle,\qquad(23)$$

in order to obtain as output a vector whose components measure the semantic relatedness of the documents in the corpora to the query.

The methods of reduction of dimensionality, "conceptualize" individual documents as weighted averages that define "pseudo documents".

To look for an explanation of the formal convergence between the biological and the technological models let us consider the following arguments:

(1) In a neural matrix memory model (Eq. 6) each input vector $\hat{f}_i$ can be the result of many association instances between N_i vectors $\hat{f}_{ij}$ and the same output vector. Thus, we can express $\hat{f}_i$ as a function of the average input $\langle f_i \rangle$ in the following way:

$$\hat{f}_i = N_i \langle f_i \rangle,$$

with

$$\hat{f}_i = \sum_{j=1}^{N_i} \hat{f}_{ij}.$$

But $\|\hat{f}_i\| f_i = N_i \langle f_i \rangle_i$, hence we can express the normalized input f_i as follows:

$$f_i = \frac{1}{N_i} \sum_{j=1}^{N_i} \left(\frac{N_i}{\|\hat{f}_i\|} \right) \hat{f}_{ij}.\qquad(24)$$

(2) It can be shown, after some calculations, that in a reduced TD-matrix the pseudo documents can be expressed by the weighted average:

$$u_h = \frac{1}{q} \sum_{j=1}^{q} \left(\frac{q}{\sigma_h} \right) v_h(j) d_j \ (h = 1, 2, \ldots, r)\qquad(25)$$

where $v_h(j)$ is the j-component of the singular vector v_h (Mizraji 2008).

In both situations, vectors are natural representations for coded data and matrices are the simplest operators connecting inputs and outputs. In addition, in the two cases the intrinsic procedures that lead to the establishment of the matrices (learning in memory models, truncated SVD in TD-matrix), generate operators that store conceptualized information. In both cases we are confronted with averaging procedures that promote the extraction of prototypes from incoming data. The contextualization procedure described at the beginning of this Section can be one of the ways to construct the neural thematic packing that the lexical human products naturally exhibit, and that the artificial search engines aim to reconstruct from text corpora.

The latent semantic correlations

Let a matrix memory $M \in R^{p \times q}$ be a device that associates normalized vectors $f \in R^p$ representing the coded version of conceptual semantic inputs, and the corresponding normal outputs $g \in R^q$. As shown in Eq. 15, these memory modules can be incorporated in complex contextual memories where the contexts produce versatile thematic clusters. Consider now the simplest structure of matrix M, given by

$$M = \sum_{i=1}^{r} \mu_i g_i f_i^T.$$

Remark that this expression -if the set $\{g_i\}$ is orthogonal- provides the SVD of matrix M and establish a direct formal contact with the TD-matrices. The direct correlation between two normalized actual input vectors f_1^* and f_2^* is given by

$$C_f(1, 2) = \langle f_1^*, f_2^* \rangle.$$

Instead, the correlation between their semantic neural interpretations is given by $C_f^*(1, 2) \equiv C_g(1, 2) = \langle Mf_1^*, Mf_2^* \rangle$.

In what follows, the correlation $C_f^*(1, 2)$ is going to be interpreted as the correlation between two new vectors δ_1 and δ_2 that map the latent semantics between inputs f_1^* and f_2^*. Remark that

$$C_f^*(1, 2) = \langle Mf_1^*, Mf_2^* \rangle = (Mf_1^*)^T (Mf_2^*) = f_1^{*T}(M^T M)f_2^*.$$

But

$$M^T M = \sum_{i=1}^{r} \mu_i^2 f_i f_i^T,$$

Hence,

$$C_f^* = \sum_{i=1}^{r} \left(\mu_i \langle f_1^*, f_i \rangle \right) \left(\mu_i \langle f_2^*, f_i \rangle \right)$$

and defining

$$\delta_1 = \begin{bmatrix} \mu_1 \langle f_1^*, f_1 \rangle \\ \vdots \\ \mu_r \langle f_1^*, f_r \rangle \end{bmatrix} \text{ and } \delta_2 = \begin{bmatrix} \mu_1 \langle f_2^*, f_1 \rangle \\ \vdots \\ \mu_r \langle f_2^*, f_r \rangle \end{bmatrix}$$

we obtain

$$C_f^* = \langle \delta_1, \delta_2 \rangle,$$

where these new vectors $\delta \in \mathrm{R}^r$ result from the confrontation between the actual inputs f^* with all the previously "conceptualized" inputs f and their corresponding frequency weights μ.

Clearly, considering the structural similarities between the theory of biological matrix memories and LSA (Mizraji 2008), the same analysis operates for the latent semantic extractions performed by LSA over artificial databases. Conversely, this illustration of the way matrix memories employ latent semantic structures, provides another point of view that helps to understand the rich "conceptualization" abilities displayed by the matrix memory models as were described many years ago by several authors (for instance by Anderson (1972) and Kohonen (1972, 1977).
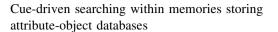
Multimodular memory systems

The evidence coming from neuroimaging and neuropsychology shows that the execution of different cognitive tasks coexists with complex dynamical dialogs between neuronal modules (see e.g., Martin et al. 1996; Raichle 2003). These neural modules are specialized neural networks prepared to perform a variety of functions, e.g., recognition of sensory inputs, translation of a visual pattern to an associated word, solving of a logical problem or a dilemma, etc. Different cognitive problems impose different trajectories between a subset of a large set of neural modules, and many of these modules are associative memories storing acquired data.

In general the Output of a module follows a general description like

$$X(t + \tau) = F(X(t), I(t), P),$$

where X is a vector that represents the state of the system (firing frequencies of neurons) including those to be considered output neurons, I is the vector of inputs, P is a vector of parameters, (e.g., synaptic weights) t is time and τ is a time scale of operation of the module. In most cases we assume that τ is a single value so all the dynamics is discrete and F can be considered as a map. In this section, we illustrate few cases of dynamic searching processes that involve multimodular nets.

Cue-driven searching within memories storing attribute-object databases

Attribute-object databases are a widespread modality for the storage of knowledge in natural and artificial systems. Examples of them are word-document databases of large text repositories and the symptoms-diseases mappings of medical knowledge. They are completely described by attribute-object tables. The main computational problem related to this way of knowledge storage, is the perfect recognition of an object of the database when partial attribute information is given.

(a) Single cue progressive searching in complete and balanced databases

A multimodular system equipped with an Attribute-Object Associative Memory M, an Intersection Filter Y and a Working Memory Module W that could maintain the last previous activity until the next cue arrives can perform a progressive narrowing in the searching space and reach, eventually, a single diagnosis (the recognition of a single object) if the set of arrived attributes suffices (Pomi and Mizraji 2001).

Imagine now a concatenated arrival of attributes a_i to the system. Each time a new attribute arrives, the memory

$$M = \sum_i \left(\sum_j s_{j(i)} \right) a_i^{\mathrm{T}}$$ provides a set of objects s_j associated

with the incoming attribute. This set of objects evoked by the current attribute, enters an intersection filter Y (as was described in Section Intersection filters, Eq. 12), to be compared with the set of objects $s(t-1)$ obtained from the history of cue attributes already presented to the system and which are retained in a short-term, working memory (Fig. 3).

The memory module Y performs the intersection between the objects compatible with the complete set of attributes arrived in the past ($s(t-1)$) and the new set of objects elicited by the last attribute that entered the system ($s(t)$). In this way, the space of possible objects is progressively pruned by the chain of incoming attributes.

In problems arising from the real world, we must deal with inconvenient variants of the aforementioned scenario, as the existence of biased databases and incomplete learning. The paramount example of this situation is medical diagnosis, in which the different prevalence of the diseases and the variable display of symptoms by individual cases induce biases in the traces stored in memory. Moreover, the extension of knowledge that the spectra of illnesses and their symptoms comprises is, by practical means, impossible to embrace, resulting in an incomplete learning of the database by any real memory.
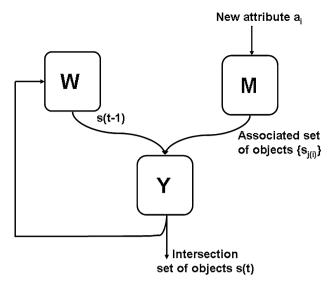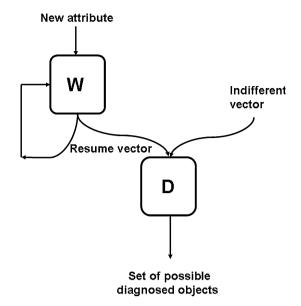
Fig. 3 A multimodular network for the progressive narrowing of a searching space. The intersection filter module Y executes the intersection between the set of objects elicited by the last attribute (via an attribute object associator M), and the set of objects saved in the working memory W, which are the result of the successive intersections between the objects elicited by each of the clue attributes presented in the previous epochs

Fig. 4 Updating the probability of different diagnoses considering the whole set of available clues. The working memory W retains the vector sum of all the attributes arrived at the system. The output of the autoassociative memory with overlapping contexts D provides, after a convenient normalization, a probabilistic map of the possible diagnoses up to time

In these situations, the output of the system (either human or machine) is usually a set of possible objects (diagnoses), varying according to the arrival of new information. A system like the one we are considering, with a biased memory, produces at each step, a weighted combination of possible diagnoses. The weights are related to the probability of each diagnosis to be the correct one, given the available symptoms. When this is the case, the unbalanced character of memory traces and the tendency to cut the low-probability diagnoses off, makes the single cue progressive searching unreliable. In fact, in certain conditions initial cues could induce the system to be captured into false pathways without return.

(b) Searching with continuous update

A partial amendment to this problem can be obtained by resetting the system and starting a new search with the whole set of available attributes every time a new cue arrives. To have an automatic neuromimetic device that behaves in this way, we must design an autoassociative memory module, D, with overlapping contexts and displace the working memory loop to the initial step (see Fig. 4). Here the working memory adds the new incoming attribute to the already available ones. The new memory module, described by the equation $D = \sum_i v_i s_i s_i^{\mathrm{T}} \otimes \left(\sum_j \alpha_{j(i)} a_{j(i)} \right)^{\mathrm{T}}$, replaces both the previous attribute-object memory M and the intersection filter Y.

As this memory is intended to learn from experience, the parameter v_i captures the frequency of presentation of object s_i and $\alpha_{j(i)}$ growths every time that the attribute $a_{j(i)}$ is accompanying this object. Each output of the system (conveniently normalized) assigns a probability to each of the possible diagnoses given the whole set of attributes that arrived to the system up to the present moment. If a new attribute arrives, it is added to the previous set of attributes within the working memory. With this aggregated vector and an indifferent vector (the sum of all possible object vectors known to the system) as the two feeding entries to the context-dependent memory D, a new evaluation is performed with no memory of past diagnoses. An associative memory system of this kind attained a good performance as an expert in medical diagnoses (Pomi and Olivera 2006), and also provides a way to explore the nature of 'cognitive bias', a known source of medical error (Pomi 2007).

Models of linguistic processing

(a) The SPELT model

One of the arenas where neural network models have been extensively used is in the area of Psycholinguistics. Among several of the models that have been created to represent linguistic faculties, the most frequently used are those based on simple recurrent networks (Elman 1990). These models have been successful in several language processing tasks

(Elman 1993) and human data fitting (Christiansen and Chater 1999), although the extent to which they capture human linguistic performance is still controversial (Marcus 2001). Simple Recurrent Networks are used as a reference model when trying to implement sequential processing in neural networks.

A particular application of these models we are interested in, is the representation of the patho-physiology of schizophrenia as excessive pruning in language processing modules (McGlashan and Hoffman 2000). The models are clearly an oversimplification of the patho-physiology of schizophrenia, but part of its predictions has been confirmed in experimental settings (Hoffman et al. 1995). The central idea behind these models is that excessive pruning of connections leads to spontaneous perception of linguistic objects in the absence of input, something that has been assimilated to verbal hallucinations. Recently we implemented a recursive variant of our context-dependent model, the Sigma-Pi Elman Topology (SPELT) model (Valle Lisboa et al. 2005). In a way resembling SRNs, the SPELT model has at its core a working memory whose output is feedback onto the same memory together with the next input.
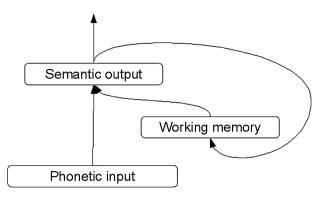
The task we model is the recognition of words in (possibly) noisy conditions. In a similar vein as Hoffman and coworkers we trained the network with an artificial language of 28 words with simple grammatical and semantic restrictions. The innovation in the SPELT is that instead of using a multilayered perceptron trained by backpropagation we employed a two layered network trained by the delta rule, using multiplicative interactions between the input and the context. In the SPELT model, the input layer feeds its output to the output layer. The output layer is copied in a working memory which projects back to the output layer as a context to the recognition of the next word. The output layer associates the Kronecker product of the input and the context with each word (Fig. 5).

In general terms, we can describe the operation of a SPELT-like model in terms of a map, by partitioning the state vector in terms of the different layers:

$$\begin{bmatrix} X_0\,(t+\tau) \\ X_{\mathrm{WM}}\,(t+\tau) \end{bmatrix} = \begin{bmatrix} M(I(t) \otimes X_{\mathrm{WM}}(t)) + \mathrm{bias} \\ X_0\,(t) \end{bmatrix},$$

being X_0 the activities of the output units and X_{WM} the activities of the working memory units, M the matrix memory and I the activity of input units. The bias term included in all units, represents a basal activity.

In all simulations performed by our group (Reali 2002) and specifically in the case of simulations analogous to those of Hoffman and coworkers (Valle Lisboa et al. 2005) the SPELT needed less training steps than SRNs to achieve good performance (over 99% recognition without noise). In

Fig. 5 SPELT model. The output layer is copied onto the working memory, this activity being used in the next time-step as the multiplicative context of the input. The interaction of both inputs is multiplicative, by means of the Kronecker product

most other respects, our model gives the same results as those of Hoffman and coworkers. For instance, the model depends on "linguistic expectation" to recognize words, for when words are presented randomly, performance drops considerably. Thus, the recurrent variant of the multiplicative model is a good alternative to more traditional approaches, one that requires less structure and training.
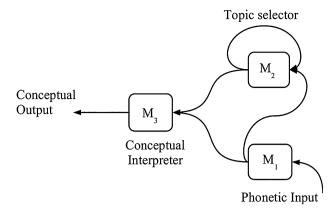
(b) Latent variables and multimodular networks

The SPELT model is a simple building block that can be used in more complex networks but has several limitations as a "standalone" module. In particular, the recognition depends only on the previous output. There is ample evidence that linguistic processing relies on different ranges of dependency, not only of syntactical but also of semantic origin (see e.g., Montemurro and Pury 2002). One possible modeling strategy for these dependencies is the inclusion of latent variables that change more slowly that the actual words of discourse, so that the observed words are attributed to a small set of conceptual frameworks. This is essentially the approach behind topic models (Blei et al. 2003; Steyvers and Griffiths 2007).

Aiming to obtain a more general model of linguistic processing we created a multimodular model with an explicit representation of topics (Fig. 6).

In its simple form, the model consists of three modules. The first one is a simple auto-associative memory that represents phonetic and phonologic processes ending in an internal representation of a word. The second module is a schematic representation of the processes whereby this initial word is used to search for information about word meaning in different modalities, usage, etc. Here we call this information a topic, and represent it as a vector. This second module associates the Kronecker product between

Topic selector

Conceptual
Output

M₃

Conceptual
Interpreter

M₂

M₁

Phonetic Input

Fig. 6 Topic-based model. Module 1 recognizes words by their phonetic or orthographic properties. The output of this module is an internal representation of the word and it is held in a working memory (not shown) for a unit of time and at the same time it enters module 2 together with the previous topic. The output of module 2 is the new topic. The new topic together with the word held in working memory are fed into the third module which assigns a meaning to the word in the context. In the absence of a topic a white vector is included as an input to module 2 (not shown)

the word and the current topic with the next topic, using the following matrix memory:

$$M_2 = \sum_{i=1}^{n} t_i \left(t_i \otimes \sum_{j=1}^{m} \Omega_j(i) w_j \right)^{\mathrm{T}}, \qquad (26)$$

where t_i represents each of the different topics (coded as orthogonal vectors), the w_js are vectors that represent the words and $\Omega_j(i)$ are weights that measure the relative importance of word j on topic i. In line with models presented in the previous section, this module acts as an intersection filter, refining the topic selection as more information is gathered.

We include a third module to disambiguate the meaning of words given the topic, which associates a word p_j and a topic t_i with a meaning c_{ij}, so each word can be associated with one meaning in each topic. This is implemented as

$$M_3 = \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} (t_i \otimes p_j)^{\mathrm{T}}.$$

(c) Word sense disambiguation: an illustration

Based on OHSUMED, a corpus of Medline abstracts (Hersh et al. 1994) we prepared a toy example using ten topics and 881 words, which is the data set we used in a previous article devoted to LSA (Valle-Lisboa and Mizraji 2007). For the simple model of Eq. 26, the presentation of three words is enough to determine the underlying topic. There are other implementation possibilities for the topic selection module, which result in different recognition capabilities (Valle Lisboa and Mizraji, in preparation). One

consequence of using the model of Eq. 26 with its multiplicative basis, is that we need to include a white context vector as a default input. The white vector consists of a linear combination of all possible topics (weighted by their frequency). If the module does not recognize any topic the next input is multiplied by the white vector, but if a topic is selected, the white vector is inhibited.

As an example, in the toy model the word INFARCTION is associated with two meanings, one related to brain infarctions (BI, topic 2, normalized weight = 0.8) and the other to heart infarctions (HI topic 3, weight = 0.2). The word CAROTID is present in both topics but with higher weight in topic 2 (weight = 0.86) than in topic 3 (weight = 0.15). On the other hand, "PRESSURE" has a higher weight in topic 3 (0.8750) than in topic 2 (0.065).

We illustrate how the model works with a simple example:

(a) When the word INFARCTION enters module 2 together with the white context, the normalized output is a linear combination of topics 2 (with a weight coefficient of 0.9701) and 3 (0.2425). If the word INFARCTION enters module 3 together with this topic the output is:

0.9701.BI + 0.2425.HI.

(b) If the word CAROTID enters first with the white context, and the resulting topic is fed-back to the module together with INFARCTION, then module 3 assigns to INFARCTION the meaning:

0.9991.BI + 0.0434.HI.

(c) If the word PRESSURE is the first word and it is followed by INFARCTION, the interpretation of INFARCTION changes to:

0.2747.BI + 0.9615.HI.

This shows that the model in its simple version can be used to achieve word sense disambiguation (Ide and Véronis 1998) a fact that deserves further exploration (Valle-Lisboa and Mizraji, in preparation).

(d) Models and modules of language processing

The SPELT and the word-disambiguating model share many properties. One of the important differences between the second model and the SPELT is that in SPELT there is only one memory block, whereas in the second model there are separate structures for topic extraction and meaning attribution. The objective of the SPELT was to show sensitivity to linguistic information in word identification. In the case of the model depicted in Fig. 6 there is more flexibility. While in the SPELT the output is also a context to a new input, in the latter model the final output can be
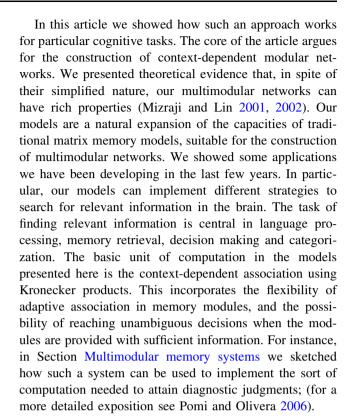
anything and that changes nothing about the next recognition. In the SPELT, there are latent associations that depend on the categorical features present in the training set, allowing the recognition of unseen combinations. The SPELT learns to associate "nounish" words as contexts to verb-like words, rendering more probable the recognition of any verb after any noun. In the multimodular model, the hidden relations are made explicit by considering a topic variable that allows for more flexibility in the effect of context, including the possibility of establishing concept associations beyond co-occurrence. This would be what is expected from models of learning that come from the information retrieval community, as LSA (Landauer and Dumais 1997), where learning new words depends on making connections between words never seen together but occurring in similar contexts.

Discussion

As we stated in the introduction, there are at least two levels at which the dynamics of the brain can be described. The first level concerns the electrochemical and biophysical processes that are displayed by the neural tissue. The other is the level of cognitive and conceptual dynamics. Of course this duality is just one phenomenon with two faces. As it happens in digital computers, where for instance the search for a file can be seen both as a complex cascade of electronic and mechanical phenomena or as a sequence of logical instructions, cognitive processes can in principle be described both at the level of action potentials or at the level of semantic units.

This viewpoint is of course related to the venerable physical-symbol system hypothesis in cognitive science (Newell and Simon 1976), but we think it is more general. In traditional cognitive science, cognitive dynamics is not open to neurobiological details, neurobiology being just an implementation of symbolic computations (Fodor and Pylyshyn 1988). Here, the dynamics of brain functioning can render subtle biophysical properties important at the cognitive level and we want to remain open to that possibility. Nevertheless, we believe that important insights can be gained by considering the dynamics of cognitive variables, much in the spirit of dynamical system approaches to cognition (van Gelder 1999), and that this approach does not need to be confined to biophysical properties.

To study cognitive dynamics, we favor a strategy based on expressing cognitive functions in the language of neural networks. This strategy requires that important pieces of psychological knowledge be implemented in neural network terms, sometimes sacrificing the biophysical details in search of a comprehensible model, which in due time can be expanded to include neurobiological detail.

In this article we showed how such an approach works for particular cognitive tasks. The core of the article argues for the construction of context-dependent modular networks. We presented theoretical evidence that, in spite of their simplified nature, our multimodular networks can have rich properties (Mizraji and Lin 2001, 2002). Our models are a natural expansion of the capacities of traditional matrix memory models, suitable for the construction of multimodular networks. We showed some applications we have been developing in the last few years. In particular, our models can implement different strategies to search for relevant information in the brain. The task of finding relevant information is central in language processing, memory retrieval, decision making and categorization. The basic unit of computation in the models presented here is the context-dependent association using Kronecker products. This incorporates the flexibility of adaptive association in memory modules, and the possibility of reaching unambiguous decisions when the modules are provided with sufficient information. For instance, in Section Multimodular memory systems we sketched how such a system can be used to implement the sort of computation needed to attain diagnostic judgments; (for a more detailed exposition see Pomi and Olivera 2006).

The algebraic simplicity of the matrix models can be exploited to understand the connection between artificial search engines and what a layer of memory neurons can do as was depicted in Section Memory modules and latent semantics. In particular, both artificial search engines and memory models use a form of information packing in terms of semantic relatedness, by means of an averaging procedure that extracts prototypes (see also, Mizraji 2008). The importance of thematic blocks evinced in these procedures (Steyvers and Griffiths 2007; Valle-Lisboa and Mizraji 2007) is instantiated in neural memories as context dependence, where information from one memory bank can modulate the associations produced in another. This does not mean that there is in fact a memory for themes or topics, since these topics could be the combined influence of information coming from different modalities that impinge in a particular module to adaptively change the output. Formally, we can assume that one or several of the modules work as a matrix memory that associates very specific words with thematic contexts, and that the context triggered by a specific word labels the subsequent flux of words until another specific word shifts the thematic context, as in the model depicted in Fig. 6 (described in Valle-Lisboa and Mizraji 2005).

To improve this approach, it can be interesting to exploit the similarities between natural and artificial procedures for meaning extraction and information retrieval from texts. The existence of those similarities between neural and computational search engine models prompts us to pursue a

research line aimed at enhancing the mutual stimulation of both fields. In such a way, technological achievements can unveil hidden neural strategies, and neural modular memories can inspire the design of innovative self-organized artificial search engines.

References

Anderson JA (1972) A simple neural network generating an interactive memory. Math Biosci 14:197–220. doi:10.1016/0025-5564(72)90075-2

Ando RK, Lee L (2001) Iterative residual rescaling: an analysis and generalization of LSI. Proceedings of the 24th SIGIR, pp 154-162

beim Graben P, Pinotsis D, Saddy D, Potthast R (2008a) Language processing with dynamic fields. Cogn Neurodyn 2:79–88

beim Graben P, Gerth S, Vasishth S (2008b) Towards dynamical system models of language-related brain potentials. Cogn Neurodyn 2:229–255

Berry MW, Browne M (2005) Understanding search engines: mathematical modelling and text retrieval, 2nd edn. SIAM, Philadelphia

Berry M, Dumais S, O'Brien G (1995) Using linear algebra for intelligent information retrieval. SIAM Rev 37:573–595. doi:10.1137/1037127

Blei DM, Ng AY, Jordan MI (2003) Latent Dirichlet allocation. J Mach Learn Res 3:993–1022

Christiansen M, Chater N (1999) Toward a connectionist model of recursion in human linguistic performance. Cogn Sci 23:157–205

Cooper LN (1973) A possible organization of animal memory and learning. In: Proceedings of the nobel symposium on collective properties of physical systems, Aspensagarden, Sweden

Cooper LN (2000) Memories and memory: a physicist's approach to the brain. Int J Mod Phys A 15(26):4069–4082. doi:10.1142/S0217751X0000272X

Deerwester S, Dumais S, Furnas G, Landauer T, Harshman R (1990) Indexing by latent semantic analysis. J Am Soc Inf Sci 41:391–407. doi:10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASI1>3.0.CO;2-9

Delmas J, Delmas A (1958) Voies et Centres Nerveux—Introduction à la Neurologie Masson, Paris

Elman J (1990) Finding structure in time. Cogn Sci 14:179–211

Elman J (1993) Learning and development in neural networks: the importance of starting small: the importance of starting small. Cognition 48:71–99. doi:10.1016/0010-0277(93)90058-4

Fodor J, Pylyshyn Z (1988) Connectionism and cognitive architecture: a critical analysis. In: Pinker S, Mehler J (eds) Connections and symbols. MIT Press, Cambridge, pp 3–71

Friston KJ (1998) Imaging neuroscience: principles or maps? Proc Natl Acad Sci USA 95:796–802. doi:10.1073/pnas.95.3.796

Graham A (1981) Kronecker products and matrix calculus with applications. Ellis Horwood, Chichester

Hersh W, Buckley C, Leone T, Hickam D (1994) Ohsumed: an interactive retrieval evaluation and new large test collection for research. Proceedings of the 17th annual ACM SIGIR Conference, pp 192–201

Hoffman RE, Rapaport J, Ameli R, McGlashan TH, Harcherik D, Servan-Schreiber D (1995) A neural network simulation of hallucinated "voices" and associated speech perception impairments in schizophrenia patients. J Cogn Neurosci 7:479–497

Ide N, Véronis J (1998) Word sense disambiguation: the state of the art. Comput linguist 24:1–41

Kandel ER, Schwartz JH (1985) Principles of neural science. Elsevier, New York

Kohonen T (1972) Correlation matrix memories. IEEE Trans Comput C 21:353–359

Kohonen T (1977) Associative memory: a system-theoretical approach. Springer, New York

Landauer T, Dumais S (1997) A solution to Plato's problem: the latent semantic analysis theory of acquisition, induction and representation of knowledge. Psychol Rev 104:211–240

Marcus G (2001) The algebraic mind. The MIT Press, Cambridge

Martin A, Wiggs CL, Ungerleider LG, Haxby JV (1996) Neural correlates of category-specific knowledge. Nature (London) 379:649–652

McGlashan TH, Hoffman RE (2000) Schizophrenia as a disorder of developmentally reduced synaptic connectivity. Arch Gen Psychiatry 57:637–648

Mizraji E (1989) Context-dependent associations in linear distributed memories. Bull Math Biol 51:195–205

Mizraji E (2008) Neural memories and search engines. Int J Gen Syst 37:715–732

Mizraji E, Lin J (1997) A dynamical approach to logical decisions. Complexity 2:56–63

Mizraji E, Lin J (2001) Fuzzy decisions in modular neural networks. Int J Bifurc Chaos 11:155–167

Mizraji E, Lin J (2002) The dynamics of logical decisions: a neural network approach. Physica D: Nonlinear Phenomena 168–169C:386–396

Mizraji E, Pomi A, Alvarez F (1994) Multiplicative contexts in associative memories. Biosystems 32:145–161

Montemurro MA, Pury PA (2002) Long-range fractal correlations in literary corpora. Fractals 10:451–461

Nass MM, Cooper LN (1975) A theory for the development of feature detecting cells in visual cortex. Biol Cybern 19:1–18

Newell A, Simon HA (1976) Computer science as empirical inquiry: symbols and search. Commun ACM 19:113–126

Papadimitriou CH, Raghavan P, Tamaki H, Vempala S (2000) Latent semantic indexing: a probabilistic analysis. J Comput Syst Sci 61:217–235

Pike R (1984) Comparison of convolution and matrix distributed memory systems for associative recall and recognition. Psychol Rev 91:281–294

Pomi A (2007) Associative memory models shed light on some mechanisms underlying cognitive errors in medical diagnosis, 6th International Conference of Biological Physics, Radisson Victoria Plaza, Montevideo

Pomi A, Mizraji E (1999) Memories in context. BioSystems 50:173–188

Pomi A, Mizraji E (2001) A cognitive architecture that solves a problem stated by Minsky. IEEE Trans Syst, Man, Cybernet-Part B: Cybernet 31:729–734

Pomi A, Mizraji E (2004) Semantic graphs and associative memories. Phys Rev E, 70, 0666136, pp 1–6

Pomi A, Olivera F (2006) Context-sensitive autoassociative memories as expert systems in medical diagnosis. BMC Med Inform Decis Mak 6:39

Raichle ME (2003) Functional brain imaging and human brain function. J Neurosci 23:3959–3962

Reali F (2002) Interacciones multiplicativas en modelos de redes neuronales: algunas aplicaciones en redes de procesamiento del lenguaje. Tesis de Maestria. PEDECIBA—Facultad de Ciencias, Uruguay

Spitzer M (1999) The mind within the net. MIT Press, Massachusetts, Chap 10

Steyvers M, Griffiths T (2007) Probabilistic topic models. In: Landauer T, McNamara DS, Dennis S, Kintsch W (eds) Handbook of latent semantic analysis. Erlbaum, Hillsdale

Tsuda I (2001) Toward an interpretation of dynamic neural activity in terms of chaotic dynamical systems. Behav Brain Sci 24(5): 793–847

Valle-Lisboa JC, Mizraji E (2005) Un modelo neuronal de procesamiento de lenguaje basado en herramientas de búsqueda de información. In Abstracts of the XI Jornadas de la Sociedad Uruguaya de Biociencias, Minas, Uruguay, 2–4 September 2005

Valle Lisboa JC, Reali F, Anastasía H, Mizraji E (2005) Elman topology with sigma-pi units: an application to the modeling of verbal hallucinations in schizophrenia. Neural Netw 18:863–877

Valle-Lisboa JC, Mizraji E (2007) The uncovering of hidden structures by latent semantic analysis. Inf Sci 177:4122–4147

van Gelder TJ (1999) Dynamic approaches to cognition. In: Wilson R, Keil F (eds) The MIT encyclopedia of cognitive sciences. MIT Press, Cambridge, pp 244–246

Wright JJ, Rennie CJ, Lees GJ, Robinson PA, Bourke PD, Chapman CL, Gordon E, Rowe DL (2004) Simulated electrocortical activity at microscopic, mesoscopic and global scales. Int J Bifurc Chaos 14:853–872