Measuring Representation

Lars Marstaller (lars.marstaller@maccs.mq.edu.au)
Macquarie Centre for Cognitive Science, Macquarie University
Sydney, NSW 2109 Australia

Arend Hintze (arend_hintze@kgi.edu)
Keck Graduate Institute for Applied Life Sciences, 535 Watson Drive
Claremont, CA 91711 USA

Christoph Adami (adami@kgi.edu)
Keck Graduate Institute for Applied Life Sciences, 535 Watson Drive
Claremont, CA 91711 USA

Abstract

We present a measure of representation in neural networks that we call '*R*', which is based on information theory. We show how *R* relates to an analysis of *distributed representation*, viz. a principal components analysis of activation space. Finally, we argue that *R* is well suited to measure representation in neural networks.

Keywords: Artificial Neural Networks; (Distributed) Representation; Information Theory; Principal Components Analysis; Measure

Introduction

Representation is a key term in cognitive science. From "good old fashioned artificial intelligence" to neural network processing, the concept of representation is used to describe and explain how cognitive systems compute (O'Brien & Opie, 2009; Pylyshyn, 1984). It is easy to apply the concept of representations to symbolic computational systems by simply mapping representations onto symbols. Rules or productions then operate on the symbolic representations to yield results. While this is a straightforward computational method, we are confronted with serious problems once we attempt to measure representation in artificial neural networks (ANNs).

ANNs do not trade in internal symbols but process information sub-symbolically (Smolensky 1987). Hence, it is very difficult to apply the division between symbols and rules. If it were possible, one should find representational vehicles, i.e. physical structures on which representational content supervenes.

We argue that instead of trying to apply the method of symbolic computation to ANNs and trying to identify clear representational vehicles, the terms should be reinterpreted. Consequently, we do not insist on the distinction between representational vehicles and computational processes. We see our measure as a first step in this direction and propose the use of information theory as a means to re-define the term representation for ANNs.

Information theory is suitable for this task because it allows to capture the meaning of representation without the need to exactly define its vehicles. As far as we understand it, representation refers to a relation between internal states of a system and some properties of its environment. Specifically such properties or states of its environment that are not actually existing (as goal states, possible outcomes of an action, or past states), not actually perceptible (because they are outside its input resolution), or not actually possible (counterfactual states) are useful in solving "representation hungry problems" (Clark 1997: 166), i.e. problems "for which a representational understanding seems *most* appropriate" (dto.). Representation then refers to "inner states that exhibit a systematic kind of coordination with a whole space of environmental contingencies" (Clark 1997: 147). We suggest that information theory and especially mutual information is able to measure this "systematic kind of coordination".

However, it is still the widely accepted assumption that the concept of representation in terms of symbolic computation can be applied to ANNs without essentially changing its meaning. One such approach is developed in the idea of distributed representations. Here it is assumed that instead of single discrete symbols, representations in ANNs consist of the activity of several processing units at once. That is, representations in ANNs are considered to be spread across several physical vehicles instead of just one.

There exist many methods to analyze neural networks in terms of distributed representation. They concentrate mainly on a principal components analysis (PCA) of the activation space of the hidden layer of a feed-forward network (see below).

Such methods have the advantage of correlating representations with regions in activation space but they are unable to provide a quantitative measure of the representational capacity of a given ANN. In order to fill this gap and to provide a new interpretation of representation, we present a measure of representation *R* that is not based on activation space partitioning but on the mutual information relations between an ANN and its world. In addition, we show how *R* is related to the concept of distributed representations in activation space.

We first review the established method of using PCA to analyze the activation space of the hidden units of ANNs. We then proceed to define our alternative measure *R* and relate it to the PCA method of identifying distributed

Article DOI: 10.5096/ASCS200935

representations in ANNs. Finally, we give an interpretation of how *R* measures representation and argue that it has the capacity to complement standard analyses of representation in ANNs.

Distributed Representations

In ANNs, computation is achieved by spreading information processing over a number of simple units. There is no central processor that, as in a Turing machine, manipulates symbol structures according to certain rules. Instead, each processing unit in an ANN transforms its input according to a mathematical function (a threshold function such as the hyperbolic tangent). The output of a unit is then sent to other units for further processing.

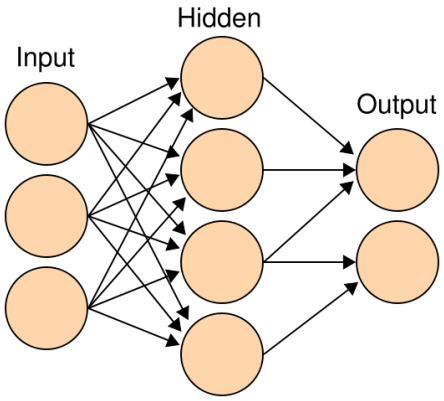

Figure 1: Artificial Neural Network

In an ANN, the global structure of the network plays a crucial role. In a network, each unit receives input from and sends output to many other units via connections with different weights. In a standard feed-forward ANN, there are usually at least three layers. Each unit of one layer is connected to all the units in the next layer (cf. Figure 1). Generally, the layered structure of the network is chosen by the researcher who determines a fit between network and task. The performance of any given network is optimized by tuning the weights of the connections. The functioning of the network itself can be analyzed by studying the activation space of the network's hidden layer, which can be viewed as the computing machinery that connects the input to the output.

Activation Space Partitioning

Activation space partitioning is performed by taking the different output values that a single unit in the ANN can produce, and render them as a vector. Taken together, multiple units constitute a vector space that represents all possible value combinations these units can assume. Each input to such a set of units occupies a specific location in this so-called activation space.

Representations are then identified as *partitions* in activation space of the hidden layer units. In order to be part of the same representation, inputs to the network must share certain properties. This sharing of properties is then reflected in the representations as a spatial property, that is, it is assumed that representational systems pick out these properties by weighing the connections between the units so that similar inputs elicit activations that are close to each other in activation space (cf. Churchland & Sejnowski, 1992). However, these activations seldom occupy identical locations in activation space simply because the inputs they represent do not share all properties, but only those that are relevant. In other words, the grouping of vectors in activation space corresponds to *relevant* properties of inputs.

In order to find the representations that are distributed over the unit value vectors, several methods are used to carve up the activation space into partitions. In general terms, partitioning activation space is the task of finding groups of activations on the basis of spatial proximity. But since a vector space of *n* units is *n*-dimensional, this space has to be projected onto a space of lower dimensions.

A common method to achieve this is principal components analysis (PCA). Sets of vectors that each encode the state of the hidden layer in response to a certain input are correlated in complicated ways. In order to find these correlations, the eigenvectors of this set are computed. Eigenvectors can be seen as the canonical form of such a set. The transformation of a set of vectors into its eigenbasis is such that if operation A is applied to an eigenvector $\vec{x}$, this vector is simply scaled by the eigenvalue λ, so that:

$$A\vec{x} = \lambda\vec{x},$$

Equation 1: Eigenvalue equation

rather than changing the vector's direction. For each eigenvector there exists a specific eigenvalue, and eigenvectors are linearly independent. PCA then is a way to analyze a set of vectors into a set of eigenvectors so that the first principal component of the set is the eigenvector with the largest eigenvalue. This means that a vector space is reduced to a set of eigenvectors that are ordered according to how much each of them accounts for the variability found in that vector space. By focusing only on the components of the initial vectors that account for the most variability, the dimensionality of the activation space of an ANN can be reduced tremendously.

Elman (1991) describes a PCA of a simple recurrent network for sentence processing. In this task, the network had to predict the next word in a sentence (or rather the context-dependent likelihood vector of the next word in a sentence; cf. Elman, 1991: 204). After applying a PCA, Elman correlates regions of the subspace that is defined by a small number of principal components with linguistic properties such as agreement, verb-argument structure, and relative clauses (Elman, 1991: 211-7).[1] As a consequence

[1] The regions are actually defined as attractor basins. That is, certain states are identified by PCA to act as attractors. The activation space can thus be partitioned on the basis of the attractor

each of these elements could reliably be correlated to particular trajectory pattern in low-dimensional principal component spaces.

In this manner, PCA allows the identification of distributed representations. As we will show, such a description is closely related to our measure of representation *R*.

Representation: The *R*-Measure

Our measure *R* contains two components: the relation between the network's internal state and the network's input, as well as the relation between the input of a network and the represented world-state. Thus, instead of giving a definition of representation in terms of the relation between a representing and a represented, we argue that – in the case of distributed representations – this relation is better explained as a higher-order relation that holds between the two components just described. More precisely, we define *R* as the *difference* between the first and the second component, viz. the difference between the relation between input and internal state, and the relation between internal state and the represented world-state (cf. Equation 3). The reason for this is that the internal states of the network are causally determined by the input states. That means that every correlation that exists between the internal states and the represented world might be due to the correlation between the input and the world states. If the correlation between the latter is subtracted from the former, we can be sure that representation is based on the relation between the internal states and the world states.

This higher-order relation also captures the difference between a camera and a representational system, for example. A camera's internal states do not qualitatively differ from its inputs, in the sense that there is a one-to-one relationship between the relevant characters of the input (such as location and color of a pixel), and the camera's state. Between the two, there is no substantial loss of information as far as human perceptual systems are concerned, that is, we can base our actions on what we see with our own eyes or by courtesy of an implanted camera.

But there is a difference between a camera and a representational system: The way a camera picture represents a visual scene is not what we are interested in when we use the concept of representation in cognitive science. What we mean by representation implies some kind of knowledge[2], some capacity to generalize from the specific object to a concept. A system that represents a particular object *knows*, in a sense, that there is this kind of object. Thus, the term representation as it is used in cognitive science refers to higher-order properties a system is able to discern over and above, e.g., the different wavelengths of the input to its visual system. The lower-order relations between the input and the internal state and between the internal state and the world-state are defined by mutual information, which we introduce below.

Mutual Information and *R*

Mutual information is an information-theoretic measure that describes the statistical dependence (correlation) of two variables on each other. If two variables are correlated, one can predict (in part or in total) the other. Thus, mutual information also measures how predictive one variable is *about* another. The mutual information between *X* and *Y* is defined as (Shannon, 1948)

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p_{xy} * \log \frac{p_{xy}}{p_x * p_y}$$

Equation 2: Mutual information of *X* and *Y*

where the random variable *X* is a set of symbols *x* that are elements of *X*, and the probability p_x defines the chance of finding the symbol *x* in the set *X*. The mutual information between *X* and *Y* (also sometimes called "shared entropy"), quantifies how well *X* can be predicted from knowing *Y*, and vice versa. In other words, it quantifies how much *X* and *Y* know about each other. For example it can easily be seen that $I(X;Y) = 0$ if *X* and *Y* are independent because then $p_{xy} = p_x * p_y$ and $\log 1 = 0$. At the other extreme, if *I(X;Y)* is maximal, then drawing *x* from the set *X* predicts which *y* will be drawn.

The relation between input (*I*) and world (*W*) and between input and internal states (*S*) can then be cast in terms of mutual information. We define our measure *R* thus as

$$R = I(S;W) - I(S;I),$$

Equation 3: Representation *R*

that is, the difference between what the internal states know about the world and what they know about what is reflected of the world in the input. Using mutual information to quantify the lower-order relations between a system's internal states and the world-states, and between internal states and input, follows our intuition about representation. Representations are used in cognitive science to describe structures that make the relevant higher-level properties of the world available for processing (cf. Clark, 1993: 87ff.; Thornton, 2009). This concept is fleshed out in the difference between *I(S;I)* and *I(S;W)*. Higher-level properties are something over and above what is immediately present in the system's input, that is, they are *generalizations* of that what is represented in the inputs. Consequently, if the system's internal states only reflect what is present in the input, the system is not representing at all, it is merely mirroring. A camera, for example, has $R=0$ because in that case, *I(S;W)* is equal to *I(S;I)*. This observation also fleshes out the above claim that a camera's internal state is not qualitatively different from its input.

landscape. For the sake of simplicity, we will use the terms region and attractor synonymously.

[2] Knowledge as understood in the sense that representations are "inner structures that act as operators upon the world via their role in determining actions" (Clark, 1997: 47).

This definition of *R* also ensures that it is compatible with the analysis of distributed representation by partitioning the network's activation space.

R and PCA

As shown above, the PCA of a network's state yields a set of independent eigenvectors describing that state. Distributed representations are encoded as regions in the space of a subset of eigenvectors. This approach is an attempt to apply the representational-computational framework to ANNs, and assumes that representations are nothing more than regions in state space. In a certain sense then, PCA categorizes activation states and labels them as representations. This is problematic because in ANNs, computation and representation cannot be easily distinguished.

Let us compare the *R* measure to PCA. If we apply PCA to the image generated on the CCD of a simple camera, we will find that it generates a number of eigenvectors that is equal to the number of inputs. So for example if the camera has 16 inputs, the PCA yields 16 eigenvectors. What is more, the eigenvectors are all equidistant to each other. This implies that the processing that the camera applies to its inputs does not in fact generate any information besides the signals on the CCD.

While a PCA of the camera states may or may not yield results that can be interpreted as a camera creating representations of the world, it appears obvious that it should not. The problem arises because the whole concept of representation, as applied to ANNs, is not rooted in the idea that representations are *about something* in the world. Instead, the identification of the relation between representing and represented is, when it comes to using PCA as gauge of representation, left to the beholder.

In contrast, our measure *R* begins with assuming that the states represented in the system are about the world, and that the concepts we form about them have to pass through the filter of the sensory system. (cf. Equation 3). Still, *R* can be thought of as encompassing PCA, because the regions in vector space that are identified by PCA are interpreted as corresponding to some property of the world. In other words, PCA attempts to create a mapping between world and regions in state space. These mappings are a set of many-to-one relations, as each region is a set of points in state space that corresponds to a number of states in the world.[3]

This is certainly the case for the network analyzed in Elman (1991) because there the number of inputs is equal to the number of entities in the world. In real world examples, where many different states of the world elicit a similar pattern on the sensory inputs, the mapping relation has to be extended. Instead of a single mapping, there exist two sets of many-to-one mappings: one from the world to the inputs, and one from the inputs to the regions. Furthermore, not all states of the world map onto the regions with the same probability (this is only the case in the camera example). Instead, for *R* the probability distribution of the mapping is included in the definition of mutual information as the joint probabilities of events p_{xy} (cf. Equation 2).

PCA identifies specific regions in activation space that are qualitatively interpreted as representations. If the activation space is partitioned so that the regions are easily distinguishable, then the system is able to perform a classification task better. In other words, in that case there is a likelihood for the states of an ANN to end up in a given region, given a set of input states that represent the same concept. This probabilistic relationship allows us to link the PCA concept of representations to that implied by *R* as follows.

The concept of activation space partitioning can be extended to information theory by assigning each possible state (be it world, input, or internal state) a particular symbol. Then, mutual information can be used to learn how much each symbol/state is predictive about another symbol/state. As a result, a region becomes interpretable as a symbol that is informative about another symbol. In the camera example, the camera would map each possible input state to exactly one internal state. So the probability of having exactly one internal state given (that is, seeing) a particular input state is $p = 1.0$. As a consequence, when the internal states have perfect information about the input states, *I(S;I)* becomes maximal. Further, if there are as many inputs as there are states in the world, then each state of the ANN automatically describes the world perfectly. In this case, $I(S;W) = I(S;I)$ because $I=W$ (the world *is* what it seems), implying a vanishing *R* – no representation.

In contrast to a camera, an ANN represents if two conditions are fulfilled: First, the possible number of input states has to be less than the number of possible world states, i.e. $I<W$. In this case, the maximum of *I(S;I)* can become smaller than the maximum of *I(S;W)* and *R* can become positive. In other words, ANNs perform a computation only if the number of possible states is reduced. Second, the internal states have to reflect some properties of the world that are not immediately accessible in the input states. An ANN can be in a state corresponding to a state of the world $w \in W$ (have information about state *w*) even though it is never reflected in the input states, i.e., $I(I;W=w)=0$. Therefore, *R* can be positive if the internal states share information with the world even though this information was never directly encoded in the input, but had to be inferred or generalized from the input.

From this it can be seen that our measure *R* includes the mappings from the regions in state space to properties in the world. If all regions in activation space and their mappings were known, it would be possible to reconstruct *R*. Thus, *R* appears to include, or is at least compatible with, the PCA of an ANN. However, the crucial difference is that *R* is a *quantitative* measure whereas state space partitioning is a

[3] Thus, the regions can be interpreted as being centered around an attractor that, semantically, is a prototype (cf. Churchland & Sejnowski, 1992: 167f.).

qualitative approach. This has further implications for the interpretation of *R*.

Interpretation of R

Critics might argue that our measure *R* measures computation instead of representation because in our definition representation is the result of a computation. We think that such arguments are based on the assumption that a clear-cut distinction between representation and computation is possible in ANNs. But we oppose this view mainly because we see a need for a re-interpretation of the terms representation and computation in ANNs.

Skeptics might worry that our measure *R* is not able to capture individual representations (that it does not measure how well individual states of the world are represented) and thus does not measure representation at all. But our formalism allows to quantify *R* for specific states:

$$R(w) = I(S; W = w) - I(S; I_w)$$

Equation 4: *R* for specific states.

where I_w is a random variable with a probability distribution given by

$$p(I_w) = p(I = i|W = w)$$

that is, it is the set of inputs that one observes when looking at the world in state *w* (cf. Adami & Cerf, 2000). With this, *R* can be calculated for any representations identified by a PCA.

Even though *R* itself does not quantify any single representation, it nonetheless captures representation because representation can be thought of as an *actualized representational capacity*. The maximal representational capacity of a system is defined as the difference between the entropy of its internal states (*S*) and the entropy of the input (*I*) (with *W* denoting world):

$$R_{max} = H(S) - H(I) \text{ with } H(W) > H(S)$$

Equation 5: Maximal Representation R_{max}

In case *R* is negative, the system has to be interpreted as relying on *external* rather than internal representations. External representations are structures that are so reliable in the system's environment that it is 'cheaper' to access them via the input than to internally represent them (cf. Brooks, 1991). This idea is reflected in our measure: R < 0 if *I(I;W) > I(S:W)*, that is, the network's input predicts the world better than its internal states. Using external representation in this way is an alternative to saying that such systems do not, in fact, realize cognitive processes. But since we assume that all systems that realize cognitive processes are to a certain degree representational, the representations have to be externally realized.

A related problem is that in order to measure *R* one has to know the states of the world that could potentially be represented. It is necessary to compute *H(W)* because the probability of events is based on their observed frequency (cf. Equation 2). In well-defined experimental situations this is not a problem. But in real world examples one would first have to define the task and the information in the world that is necessary to accomplish it. But this presupposes that one can define the minimal system that is necessary to solve the task. But the task might have many possible solutions. However, an estimate can be given for most situations. The only caveat is that one shouldn't assume that one is always in a position to control the design space (cf. Clark, 1997: 160).

Conclusion

We have shown how *R,* a new quantitative measure of representation*,* can be related to a widely used approach to distributed representations (PCA), and possibly encompasses it. In contrast to PCA, *R* is a quantitative measure that allows us to gauge the actualized representational capacity of an ANN *precisely*. It does so without identifying individual representational vehicles and thus does not try to apply a specific conceptual-explanatory framework to the analysis of ANNs. Rather, *R* is compatible with other approaches especially those that are based on information theory and dynamical systems theory. Further, by providing a quantitative and *computable* measure, *R* can be used to compare ANNs and even assess their computational efficiency.

References

Adami, C. & Cerf, N. J. (2000) Physical complexity of symbolic sequences. *Physica D, 120*.

Brooks, R. A. (1991). Intelligence without representation. *Artificial Intelligence, 47*, 139–159.

Churchland, P. & Sejnowski, T. J. (1992). *The computational brain*. Cambridge, MA: MIT Press.

Clark, A. (1993) Associative engines. Connectionism, concepts, and representational change. Cambridge, MA: MIT Press.

Clark, A. (1997). *Being there: putting brain, body, and world together again*. Cambridge, MA: MIT Press.

Elman, J. L. (1991). Distributed representations, simple recurrent networks, and grammatical structure. *Machine Learning, 7*, 195-225.

O'Brien, G. & Opie, J. (2009). The role of representation in computation. *Cognitive Processing, 10*, 53-62.

Pylyshyn, Z. (1984). *Computation and cognition: toward a foundation for cognitive science*. Cambridge, MA: MIT Press.

Shannon, C. E. (1948) A mathematical theory of communication. *Bell System Technical Journal, 27*, 379–423 & 623–656.

Smolensky, P. (1987). Connectionist AI, symbolic AI, and the brain. *Artificial Intelligence Review, 1*, 95-109.

Thornton, C. (2009) Representation recovers information. *Cognitive Science 33(8),* 1-30.

Figure 1 from: http://en.wikipedia.org/wiki/File: Artificial_neural_network.svg (under the GNU FDL)

Citation details for this article:

Marstaller, L., Hintze, A., Adami, C. (2010). Measuring representation. In W. Christensen, E. Schier, and J. Sutton (Eds.), *ASCS09: Proceedings of the 9th Conference of the Australasian Society for Cognitive Science* (pp. 232-237). Sydney: Macquarie Centre for Cognitive Science.
DOI: 10.5096/ASCS200935
URL:
http://www.maccs.mq.edu.au/news/conferences/2009/ASCS2009/html/marstaller.html