Next Word Prediction in a Connectionist Distributed Representation System

João Luís Garcia Rosa
Mestrado em Sistemas de Computação - PUC-Campinas
Mestrado em Informática - UniSantos
Rodovia D. Pedro I, km. 136 – Caixa Postal 317 – CEP 13012-970 – Campinas, SP, Brasil
joaol@ii.puc-campinas.br – Fax: +55-19-3756-7195

***Abstract*—Connectionist natural language processing models that consider the temporal extension of sentence analysis often make use of local representation, allocating only one unit for each word at the input and output layers of the connectionist architecture. Thus, for increasing the lexicon, it is mandatory to modify the architecture and re-train the network. On the other hand, the proposed system *Pred-DR* attempts to predict the next word in declarative sentences presented sequentially one word at a time, giving meaning to the units of the connectionist architecture by means of distributed representations based on semantic features. The words are fractionated into their semantic microfeature arrays. Consequently, *Pred-DR* is able to generalize to new words without increasing the number of processors in its architecture, provided that their semantic features are supplied. This way, it is achieved a considerable performance on connectionist natural language processing using the classical semantic microfeature framework. The system learns to relate the input word array to its possible next word, "remembering" the previous words seen before in a semantically sound sentence. For each input word, *Pred-DR* gives, as outcome, a list of probabilities of occurrence of next words in the sentence context.**

***Keywords*: natural language processing, neural networks, distributed representation, word prediction**

I. Introduction

Pred-DR is a connectionist system designed to predict the next word in declarative sentences through a partially recurrent neural network inspired by Elman (1990). This kind of word prediction is argued to be relevant to Natural Language Processing (Rohde 2002). Its architecture is composed by four layers: the input layer, where the words of a sentence are input one at a time, in terms of their semantic microfeature distributed representations (McClelland and Kawamoto 1986); the hidden layer, designed to develop internal distributed representations, as usual; the output layer, from which the next word (in terms of semantic features) in the sentence context is supposed to be predicted; and the context layer, which has the same size of the hidden layer and, in an Elman standpoint (Elman 1990), is able to contribute to the propagation of the input signal to the hidden layer, storing the last state of the network representation, which means that to the system is given memory. *Pred-DR* stands for *Prediction of the next word in a connectionist language processor through Distributed Representations*.

Unlike McClelland and Kawamoto (1986)'s system, *Pred-DR* has only one network for all verbs, and contrasting Rohde and Plaut (1999)'s system, it uses a distributed representation for the words at the input and output layers. This is very interesting in a psycholinguistic standpoint, because this way *Pred-DR* is able to generalize over verbs and nouns and even for a word not present in its lexicon, it is possible for the system to incorporate it, only supplying its semantic feature array.

II. Distributed Representations

Several are the advantages of the distributed representation concerning connectionism. According to Hinton and others (Hinton, McClelland, and Rumelhart 1986),

> "the connections between a set of simple processing units are capable of supporting a large number of different patterns."

This implies in a considerable reduction of the network size. And, regarding cognition,

> "the strengths and weaknesses (of the distributed representations) match those of the human mind"

and

> "give rise to some powerful and unexpected emergent properties, (like) generalization."

This way, systems that employ distributed representations are more "psycholinguistically realistic". Another point concerns generalization:

> "generalization is normally a helpful phenomenon, to deal effectively with situations that are similar but not identical to previously experienced situations."

Distributed representations

> "make it possible to create new concepts without allocating new hardware."

This means that new words can be added to the lexicon of systems that use distributed representations like *Pred-DR*, without modifying the architecture previously employed and trained. This is not true for systems with local representations (for instance, Elman 1993; Rohde and Plaut 1999).

A. Semantic Microfeatures

A classical approach for distributed representations is the semantic feature encoding, used by Waltz and Pollack (1985) and by McClelland and Kawamoto (1986). This kind of representation is meaningful by itself. It is possible to extract information just by examining the representation, and different systems can process the same representations and communicate using them. In addition, since these representations are semantically well constructed, they may be related to a semantic theory (for instance, the Leech's Semantics (Leech 1974)).

On the other hand, such patterns must be preencoded and they remain fixed. Because all the concepts must be classified along the same dimensions, the number of dimensions may become very large, and many of them may be irrelevant to a particular concept. It is very hard to decide what dimensions are necessary and useful for a given problem (van Gelder 1989).

There is also the epistemological question of whether the process of deciding what dimensions to use is justifiable or not. Hand-coded representations are always more or less ad hoc and biased. In some cases, it is possible make the task trivial by a clever encoding of the input representations (Miikkulainen 1993).

In *Pred-DR*, word representation is adapted from the classical distributed semantic microfeature representations used by McClelland and Kawamoto (1986), for nouns. For verbs, *Pred-DR* uses the representation employed in systems HTRP (Rosa and Françozo 1999) and HTRP-II (Rosa 2001). Twenty three-valued logic semantic microfeature units account for each noun and verb. The schema on table 1 displays the semantic features for verbs. Table 2 shows the microfeatures for nouns.

Table 1. The ten semantic microfeature dimensions for verbs

control of action	no control of action
direct process triggering	indirect triggering
direction to source	direction to goal
impacting process	no impacting process
change of state	no change of state
psychological state	no psychological state
objective	no objective
effective action	no effective action
high intensity of action	low intensity
interest on process	no interest on process

It is important to notice here that the verb microfeatures are chosen in order to encompass the semantic issues considered relevant in a semantic role frame. The microfeatures outside this context are not purposeful (Rosa and Françozo 1999; Rosa 2001).

Table 3 shows two verbs used in *Pred-DR* with their semantic microfeatures, as an example. Table 4 displays some nouns.

Table 2. The seven semantic microfeature dimensions for nouns, separated in rows. Only one value in each dimension is on for each unambiguous noun (adapted from McClelland and Kawamoto 1986)

human			non-human		
soft			hard		
small		medium		large	
1-D/compact		2-D		3-D	
pointed			rounded		
fragile/breakable			unbreakable		
value	furniture	food	toy	tool/ utensil	animat e

Table 3. *Pred-DR* verb microfeatures for two verbs (*break* and *fear*), in terms of symbolic expressions. For ambiguous verbs there are two possible readings, for instance, *break1* and *break2*. In this case, the "?" stands for unknown value for the default reading. See table 1

microfeature	*break*	*break1*	*break2*	*fear*
control of action	?	no	yes	no
process triggering	?	indirect	direct	indirect
direction	goal	goal	goal	source
impacting process	yes	yes	yes	yes
change of state	yes	yes	yes	no
psychological state	no	no	no	yes
objective action	?	no	yes	no
effective action	yes	yes	yes	no
intensity of action	high	high	high	low
interest on process	?	no	yes	no

III. The Lexicon

The lexicon used in *Pred-DR* includes ambiguous nouns and verbs. For instance, in relation to nouns, lexically ambiguous words as *chicken* are included. The system is able to decide which *chicken* is intended, considering the whole sentence in which *chicken* occur as context, like the disambiguation resource presented in McClelland and Kawamoto (1986)'s system. For verbs, there are different verbs and "alternative" readings of a same verb, concerning a kind of "semantic-role" ambiguity. That is, some verbs, for instance *break*, may have three operands, like in sentence (1), or two operands like in sentence (2). This implies a different *thematic role* assignment by the verb to these operands. The discussion of how semantic roles are important and influence sentence compositionality is presented in Rosa and Françozo (1999) and in Rosa (2001).

Table 4. *Pred-DR* microfeatures for some nouns, in terms of symbolic expressions (adapted from McClelland and Kawamoto 1986). See table 2

Noun	**human**	**softness**	**volume**	**form**	**pointness**	**breakability**	**object type**
ball	non-human	soft	small	3-D	rounded	unbreakable	toy
boy	human	soft	medium	3-D	rounded	unbreakable	animate
chicken	non-human	soft	medium	3-D	rounded	unbreakable	?
chicken (food)	non-human	soft	medium	3-D	rounded	unbreakable	food
chicken(animal)	non-human	soft	medium	3-D	rounded	unbreakable	animate
desk	non-human	hard	big	3-D	pointed	breakable	furniture
dog	non-human	soft	medium	3-D	rounded	unbreakable	animate
girl	human	soft	medium	3-D	rounded	unbreakable	animate
hammer	non-human	hard	small	compact	pointed	breakable	tool/utensil
man	human	soft	big	3-D	rounded	unbreakable	animate
monkey	non-human	soft	small	3-D	rounded	unbreakable	animate
spaghetti	non-human	soft	small	compact	rounded	breakable	food
spoon	non-human	hard	small	compact	pointed	breakable	tool/utensil
stone	non-human	hard	small	3-D	pointed	unbreakable	tool/utensil
ten	non-human	soft	small	compact	pointed	unbreakable	value
vase	non-human	hard	small	compact	rounded	breakable	tool/utensil
window	non-human	hard	medium	2-D	pointed	breakable	tool/utensil
wolf	non-human	soft	medium	3-D	rounded	unbreakable	animate
woman	human	soft	big	3-D	rounded	unbreakable	animate

(1) The boy broke the window with the stone
(2) The woman broke the vase.

As a matter of fact, *break* may have only one operand, like in sentence (3), but in this case, the system does not have a third reading for *break*; instead it is treated in the two-operand reading available.

(3) The window broke.

The system Pred-DR allows the user to verify the semantic microfeatures of the words. It is possible to examine the microfeatures of the entire lexicon or to enter a specific word (noun or verb). For instance, if one enters the verb *fear*, the system gives the following, as shown on table 3:

Verb: FEAR
no control of action
indirect process triggering
direction to source
impacting process
no change of state
psychological state
no objective
no effective action
low intensity of action
no interest on process

In the case of an "alternative" reading of an ambiguous verb, for instance the verb *break*, the system gives:

Verb: BREAK
? control of action
? no control of action
? direct process triggering
? indirect process triggering
direction to goal
impacting process
change of state
no psychological state
? objective
? no objective
effective action
high intensity of action
? interest on process
? no interest on process

Note that the "?" sign indicates that there is ambiguity regarding the subsequent microfeature. Resembling the noun ambiguity resolution, *Pred-DR* is able to learn the correct reading of the verb entered, based on the sentence context.

IV. The Connectionist Architecture

A former version of *Pred-DR* included a connectionist architecture with 986 locally distributed input units and 43 local output units. The input units were responsible for the representation of 42 words, with 24 semantic features each one. The output units represented the 42 words plus the *end-of-sentence marker*.

The architecture employed is a multi-layer perceptron, within a partially recurrent Elman network (Elman 1990). The representation of inputs was locally distributed, i. e., it was local, since the same set of units represents the same word, and it was distributed within each word, since semantic microfeature representation was employed. In fact, the previous version adapted the local representation used by Elman (1993) and by Rohde and Plaut (1999) to the microfeature frame representation of McClelland and

Kawamoto (1986).

The present version consists of a partially recurrent neural network also, like Elman's (Elman 1990; Elman 1993) with only 80 input units (although the network input receives one word at a time, there are specific localized set of units representing a syntactic category: a subject, the verb, an object, and a complement, with 20 semantic microfeatures each). The reason for this terrific reduction on the input layer size is that this way it is possible to generalize over verbs and nouns. The output layer has 61 units, 20 for the verb, 20 for the object, 20 for the complement and one for the end-of-sentence marker. The subject output is unnecessary because the next word in a declarative sentence will never be a subject. In addition, the sequence imposed to the network input (subject – verb – object – complement) shows that syntactic constraints are also included (figure 1).

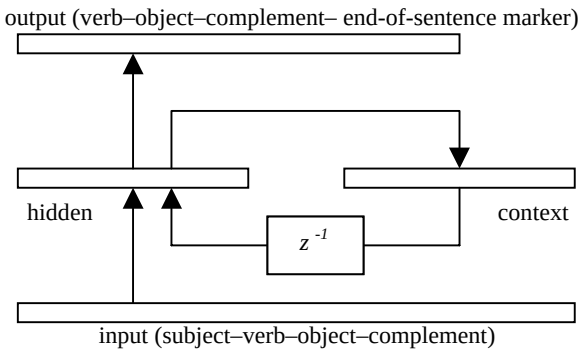

Figure 1. *Pred-DR* architecture with four layers: input, hidden, output, and an extra (context) layer in a partially recurrent Elman network. The network input receives one word at a time in the localized slot concerning the specific syntactic category.

Other words of the sentence, like articles, adjectives, adverbs and so on, are supposed to be discharged after a pre-processing symbolic module. For *Pred-DR* purposes, the sentence comes in a "canonical form", with only verb and nouns. It is possible to improve the system to account for more complex sentences using this same type of representation, providing that the network architecture follows this enhancement.

A. The Recurrence

The contribution of the context layer, which provides memory to the system, is made on the one-to-one basis; that is, each hidden unit is copied to its corresponding in context layer. In the first activation, during the training step, the context units are all set to zero (Kröse and van der Smagt 1996). At the time *t+1*, the context layer contribution is as following: (a) first, the input units are propagated to the hidden units, giving a first "hidden output"; (b) then, the context units are propagated to the hidden units, giving a second "hidden output"; (c) the average of these two "hidden outputs" will give the final "hidden output" which will operate as "input" to the output layer. Notice the unit delay operator z^{-1} in figure 1, which indicates that the context layer will contribute to the network propagation only in the next activation cycle. The training algorithm *backpropagation* (Rumelhart, Hinton, and Williams 1986) will correct the weights between the output and hidden layers and between the hidden and input layers. The connections between hidden and context units are one to one; that is, there is a copy unit per unit from the hidden layer to the context layer. The weights between the context layer and the hidden layer, through the unit delay operator, are all set to 0.1 and remain fixed. This kind of network is called *simple recurrent network* (Haykin 1999).

V. LEARNING

Pred-DR employs a sentence generator for the training step. Instead of entering the sentences by hand, they are generated automatically by a seven-frame set for each one of the verbs (including alternative readings, like the verb *break*). Recall that the sentences are presented to the network input one word at a time. And the system expected, as outcome, the next word in the given sentence. See some generating frames on table 5. The generator replaces the categories present in frames by the words for each category given on table 6.

The sentence generator supplies the training sentences, according to semantic and syntactic constraints. After about 24,000 training cycles, which corresponds to an average output error [1] of 10^{-3}, the system is able to predict which are the candidates for being the next word in a declarative sentence and their probabilities of occurrence.

When an output arises from the propagation of a word (e.g., a subject) through the connectionist architecture, the system compares this distributed representation output array (in this case, of a verb) dimension by dimension, that is, for each dimension, for instance, *control of action*, which features are equal to the expected verb dimension and which are not. This way, it is possible for *Pred-DR* to predict, after checking all the dimensions, how much the actual output array is closer to a specific word. Then, the "probability" of occurrence is given, based on the distance between an average of *active* outputs, that is, outputs that have values greater that 0.5, and the word itself.

For instance, in the sentence (4), when the subject *the boy* is inserted into the system, some verbs are highlighted as possible candidates to be the next word in a sentence with *the boy* as subject, as shown in (5).

[1] The average output error is the difference between "actual" output and "desired" output, and it is obtained from the *average squared error energy* formula (Haykin 1999).

Table 5. The frames for some verbs (break and fear) of the sentence generator in *Pred-DR*

	sentence frames for break1
1	the object broke the fragile_object
2	the breaker broke fragile_object
3	the object broke the fragile_object
4	the fragile_object broke
5	the object broke the fragile_object
6	the breaker broke the fragile_object
7	the fragile_object broke
	sentence frames for break2
1	the human broke the fragile_object with the breaker
2	the animal broke the fragile_object
3	the human broke the fragile_object with the breaker
4	the human broke the fragile_object
5	the animal broke the fragile_object
6	the human broke the fragile_object with the breaker
7	the human broke the fragile_object with the breaker
	sentence frames for fear
1	the human fears the human
2	the human fears the animal
3	the animal fears the human
4	the animal fears the animal
5	the animal fears the predator
6	the animal fears the animal
7	the human fears the human

Table 6. The categories for some frames of sentence generator (table 5)

category	***noun 1***	***noun 2***	***noun 3***	***noun 4***
animal	*chicken*	*dog*	*wolf*	*monkey*
breaker	*ball*	*hammer*	*vase*	*stone*
fragile_ object	*window*	*vase*	*plate*	*window*
human	*man*	*girl*	*boy*	*woman*
object	*ball*	*jack*	*doll*	*plate*
predator	*wolf*	*dog*	*wolf*	*dog*

(4) The boy fears the wolf

(5) **fear: 64.0%**
love: 64.0%
frighten: 42.7%
...

This means that the network output is closer to *fear* and *love* (64%) than to other verbs (*frighten*, etc.). Notice that at this time, no noun appears as possible next word (all of them display 0.0%).

When *fears* is input, the system "remember" the last word seen (*boy*). Actually, *Pred-DR* has an internal representation of boy stored in the context units, so it is able to deduce the next word in relation to the phrase *the boy fears* and not only in relation to the word *fears*. And the result is shown in (6).

(6) chicken: 91.3%
monkey: 91.3%
dog: 82.1%
wolf: 82.1%
...

Now, only nouns show importance. And finally, when the network takes *the wolf* as its input, it will display the *end-of-sentence marker* as the next element, indicating that the sentence finishes.

Another example is shown in (7). For this sentence, when *the stone* is inserted, the system displays the "prediction" of the next word as shown in (8). And when the system have already seen *the stone broke*, it tries to predict the object in this sentence (9). And, finally, *Pred-DR* indicates the end of the sentence, when the user enters *the window*, signaling that there is no complement in this sentence (it is unlikely to expect that a stone could use a tool in the act of breaking).

(7) The stone broke the window

(8) hit: 91.7%
break: 82.6%
frighten: 82.6%
...

(9) plate: 72.4%
vase: 72.4%
window: 64.4%
...

It is important to notice here that the learning ability displayed by *Pred-DR* reflects what it sees during training. Check out the training sentences for verbs *break* and *fear* on tables 5 and 6.

VI. Conclusion

Pred-DR is a partially recurrent connectionist approach to natural language processing. It employs the distributed representation for inputs and outputs, unlike known systems (such as Elman 1993 and Rohde and Plaut 1999).

Also, unlike McClelland and Kawamoto's (1986) system, in *Pred-DR* a single network accounts for all sentences; thus generalizing over *both* nouns and verbs. The distributed representations as well as the single network allow the inclusion of new words in *Pred-DR*, without allocating more hardware, provided that their semantic microfeature arrays are supplied.

Two are the main differences between the previous version of the system and the final *Pred-DR*. First, the former

version employed a connectionist network with 986 input units and 43 output units. The input units were responsible for the representation of 42 words, with 24 semantic features each one. The output units represented the 42 words plus the end-of-sentence marker. The proposed system uses one connectionist architecture with 80 input units and 61 output units, to account for four words with twenty microfeatures each as its input and three words, with twenty microfeatures each also, plus the end-of-sentence marker as the output. The other difference concerns the way the system learns: in the previous version the output is local, i. e., each unit was responsible for one entire word. In *Pred-DR*, both the input and the output are distributed representations of the words. This way, the system is able to generalize over nouns and verbs, and making possible to add new words to the lexicon, providing that their semantic microfeatures are supplied, without allocating more hardware. This means that the system would not have to be re-trained.

Acknowledgement

I would like to thank Fernando José Rossi and Hennry Lieggio Duro, computing engineering students, for the implementation and tests of the first version of *Pred-DR*.

VII. REFERENCES

[1] Elman, J. L. 1990. Finding Structure in Time. *Cognitive Science* 14, pp. 179-211.

[2] Elman, J. L. 1993. Learning and Development in Neural Networks: the Importance of Starting Small. *Cognition* 48, pp. 71-99.

[3] Haykin, S. 1999. *Neural Networks - A Comprehensive Foundation*, 2nd edition. Prentice Hall, Upper Saddle River, New Jersey.

[4] Hinton, G. E.; McClelland, J. L.; and Rumelhart, D. E. 1986. Distributed Representations. In D. E. Rumelhart and J. L. McClelland (Eds.), *Parallel Distributed Processing – Explorations in the Microstructure of Cognition, Volume 1 – Foundations*. A Bradford Book, MIT Press, pp. 77-109.

[5] Kröse, B., and van der Smagt, P. 1996. *An Introduction to Neural Networks*. Eighth edition. The University of Amsterdam. November.

[6] Leech, G. 1974. *Semantics*. Penguin Books.

[7] McClelland, J. L., and Kawamoto, A. H. 1986. Mechanisms of Sentence Processing: Assigning Roles to Constituents of Sentences. In J. L. McClelland and D. E. Rumelhart (Eds.), *Parallel Distributed Processing – Explorations in the Microstructure of Cognition – Volume 2: Psychological and Biological Models*. A Bradford Book, MIT Press, pp. 272-325.

[8] Miikkulainen, R. 1993. *Subsymbolic Natural Language Processing - An Integrated Model of Scripts, Lexicon, and Memory*. A Bradford Book. The MIT Press.

[9] Rohde, D. L. T. 2002. *A Connectionist Model of Sentence Comprehension and Production*. Unpublished Ph.D. Thesis. Computer Science Department. School of Computer Science. Carnegie Mellon University. Pittsburgh, PA.

[10] Rohde, D. L. T., and Plaut, D. C. 1999. Language Acquisition in the Absence of Explicit Negative Evidence: How Important is Starting Small? *Cognition* 72, pp. 67-109.

[11] Rosa, J. L. G., and Françozo, E. 1999. Hybrid Thematic Role Processor: Symbolic Linguistic Relations Revised by Connectionist Learning. In *Proceedings of IJCAI'99 – Sixteenth International Joint Conference on Artificial Intelligence*, Volume 2, Stockholm, Sweden, 31 July-6 August, pp. 852-857. Morgan Kaufmann.

[12] Rosa, J. L. G. 2001. HTRP II: Learning thematic relations from semantically sound sentences, in *Proceedings of the 2001 IEEE International Conference on Systems, Man, and Cybernetics - SMC2001*, October 7-10, 2001, Tucson, Arizona, United States of America, pp. 488-493.

[13] Rumelhart, D. E.; Hinton, G. E.; and Williams, R. J. 1986. Learning Internal Representations by Error Propagation. In D. E. Rumelhart and J. L. McClelland (Eds.), *Parallel Distributed Processing – Explorations in the Microstructure of Cognition, Volume 1 - Foundations*. A Bradford Book, MIT Press, pp. 318-362.

[14] van Gelder, T. 1989. *Distributed Representation*. Ph.D. Thesis, Department of Philosophy, University of Pittsburgh, Pittsburgh, PA.

[15] Waltz, D. L., and Pollack, J. B. 1985. Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretations. *Cognitive Science* 9, pp. 51-74.