11

A connectionist model for the interpretation of metaphors

Stefan Wermter
FB Informatik
Universität Hamburg

Ruth Hannuschka
Universität Dortmund

11.1 Introduction

The interpretation of non-literal metaphorical language is a very difficult but important problem in natural language processing. Metaphors are concepts which are transferred to the context of other different concepts, as for instance in 'green'[1] lawyer' or 'life is a highway'. In order to find a semantic interpretation for a metaphorical concept in a new context a metaphorical mapping between concepts has to be determined [Carbonell 1982]. In purely symbolic approaches, abstraction hierarchies have often been used to identify such a mapping between concepts (e.g. [Fass 1988], [Martin 1988] and [Way 1991]). However, current symbolic systems have two main disadvantages for metaphor interpretation. First, the symbolic knowledge has to be encoded rather than learned, and, second, the symbolic knowledge does not directly support a graded plausible representation, which is particularly important for representing a mapping for a metaphorical interpretation. In contrast, connectionist networks are able to learn similarity mappings, to generalize from known learned to unknown examples, and to support a plausible interpretation [Wermter 1989]. These properties will be exploited to build a connectionist learning-based model for metaphor interpretation.

[1] Here we mean 'green' in the sense of inexperienced.

As a new approach we describe an implemented modular model which uses connectionist representations for interpreting metaphorical language. We focus on a class of adjective–noun and noun–preposition–noun metaphors which have been taken directly from the Treebank corpus at the University of Pennsylvannia; examples are 'thirsty airplane' and 'engine in person'. This class of metaphors has been chosen since such constructions occur particularly often. In this approach metaphorical interpretations are represented in a connectionist function network by mapping source concepts onto target relationships. The influence of context on the interpretation of metaphorical language is considered by using an additional connectionist context network which controls the function network. In contrast to approaches that use an in-depth understanding and just cover relatively few metaphorical examples with a detailed encoded knowledge representation, we use an approach of a scanning understanding [Wermter 1992] which covers many examples with a shallower, more plausible representation. We will describe the learning of metaphors, analyse the connectionist representation, and point out the potential advantages of a connectionist approach with respect to performance and representation.

11.2 Analysis of existing approaches

11.2.1 Symbolic approaches

The structure-mapping theory proposed by Gentner is a well-known approach to the interpretation of analogical metaphors using an underlying conceptual *analogy* between the concept representing the literal source meaning of a word and the concept representing the target meaning of the metaphor [Gentner 1983, Gentner *et al.* 1989]. An analogy makes *explicit* the basic relationship between the source and target domains, as in the example 'The atom is like the solar system' [Gentner 1983, p. 160]. Analogies are not typically motivated from common conceptual grounding. Rather they often employ novel background mappings and therefore explicitly mark them with relationships such as 'is like'. Gentner's theory is formulated as a set of such rules for the interpretation of analogies. These rules are based on *relational* similarities rather than similarities of the underlying objects in the source and target domains.

While analogical metaphors contain explicitly marked relationships ('is like') many other metaphors do not. Word-sense approaches like that of Small and Rieger [Small and Rieger 1982] address such metaphors. In this approach a non-literal use of words is recognized and represented as separate lexical items. However, this representation excludes the possibility of generalization, so that there is no way of supporting the interpretation of one metaphor by using knowledge about another metaphor. [Wilks 1975] describes an approach that analyses natural language using a preference

selection of semantic structures. A word sense is represented by a semantic formula which describes preferences for the objects involved: for example, a 'drink action' prefers an 'animated' subject, a 'fluid' object. This approach covers metaphorical language by special mechanisms which relax the constraints on preferences, but it does not prevent the acceptance of anomalous language and it is completely hand-coded.

These approaches were criticized by Lakoff and Johnson [Lakoff and Johnson 1980, Lakoff 1987] who argued that metaphors are a conventional part of language and should not be handled differently from literal language. Lakoff and Johnson suggested a set of metaphors which they state to be basic within the language, such as 'more-is-up' and 'health-is-up'. They claimed that other more specific metaphors can be mapped onto this set of basic metaphors: for example, 'peak of health' is mapped onto 'health-is-up'. In general, this theoretical approach is interesting since it does not have the problem of distinguishing literal from non-literal language.

[Carbonell 1982] proposed an approach to metaphor interpretation which considered this notion of basic metaphors. This approach derives the literal interpretation and calls a metaphor interpretation system if a violation of the semantic case roles is detected. A recognition network identifies whether a given construct is a typical instantiation of a basic metaphor. For example 'soaring gold market' is identified as an instance of the basic 'more-is-up' metaphor by exploiting the fact that the 'upward movement' of the source concept 'soaring' matches 'up' and that the 'increase of a quantifiable property' of 'gold market' matches 'more'. The inferences that can be drawn in the source domain are applied to the target domain, and a filter mechanism selects those inferences which are valid inferences in the target domain; for instance, 'the gold market may suffer significant state change' is a valid inference from the above example. If the inferences drawn are inconsistent with the actual target or no mapping can be determined, a matching algorithm is used to search for similarities within the given source and target concepts. If the search is successful, the new mapping is stored. Carbonell proposes that the instantiations of the basic metaphors be organized in a hierarchy by degree of specialization to exploit knowledge about their relationship but that the inclusion of generalized metaphors has to be restricted to keep the system tractable.

There are other recent approaches to metaphor interpretation that use abstraction hierarchies to represent the necessary knowledge (e.g. [Zernik 1987], [Fass 1988] and [Martin 1988]). Zernik describes an approach for learning of non-literal phrases. Basic to this approach is a hierarchically organized phrasal lexicon and mechanisms for generalization and specialization of existing phrases. These features enable the system to store new phrases. The representation of phrases includes a presupposition and the interpretation. Instead of representing phrases independently from each other, common features are extracted and generalized phrases are

represented. In the absence of specific context a general phrase is applied and a specialization mechanism produces a representation of the new phrase. Discriminating conditions are needed to prevent undesired generalization. For example, 'to act around' with specifications, 'to push around' and 'to boss around' would also accept 'to stick around' in the absence discriminating conditions. However, the successful interpretation of new phrases within the system largely depends on the initial structure of the lexicon.

Similarly the work of [Fass 1988] on non-literal language benefits from the use of abstraction hierarchies. His system is a semantic network of individual word senses represented by sense-frames which determine preferences for the involved objects. Metaphorical relationships are recognized by detecting a preference violation and the existence of an analogical match within the arguments of the sense-frames. Semantic vectors specify the degree of the matching between the sense-frames, the distance within the hierarchy, and the resolution of lexical ambiguities. The system benefits from the extensive amount of encoded knowledge.

Based on the results of [Lakoff and Johnson 1980], [Martin 1988] assumed that processing metaphors involves the application of pre-stored knowledge structures representing basic metaphors. His system can apply known knowledge structures to metaphorical language and can learn new metaphors. The learning mechanism works basically by analogy mapping using a concept hierarchy. The concepts are related to each other by links like dominance and instance. The system can deal with complex source and target concepts by combining the single association between source and target concepts to a structured association representing a complex metaphor. Conceptual closeness of source concepts is taken into account, since similar concepts are described by a small distance within the conceptual hierarchy. The representation considers metaphorical coherence, that is, metaphorical mappings are themselves organized as concepts in the hierarchy. This enables the system to capture not only similarities between objects but similarities within the set of metaphorical mappings as well. More abstract metaphors are exploited to interpret special ones using special inference mechanisms for specialization similar to Zernik's approach. Martin's approach depends highly on the basic metaphors represented in the system. Careful selection of basic metaphors is needed for this approach, especially for the ability of the system to generalize. Furthermore, the method for determining the source and target concepts and the similarity measure does not particularly support the interpretation of phrases in different contexts.

11.2.2 Connectionist approaches

Recently there have been attempts to exploit connectionist representations for the interpretation of metaphors. Garrison Cottrell models word-sense

disambiguation as a constraint relaxation process [Cottrell and Small 1983, Cottrell 1988]. Knowledge is represented within nodes of a localist network and constraints between different hypotheses are described by positive and negative connections between the nodes. Cottrell suggests that the different stable states of the connectionist network could reflect related meanings and could account for the interpretation of metaphors. However, each meaning of a word is represented by a single node and its connections within the network, which makes the disambiguation and metaphor interpretation intractable for large knowledge bases.

Hollbach-Weber has developed an approach to handle the interpretation of metaphorical adjective–noun constructs [Hollbach-Weber 1989, Hollbach-Weber 1991]. Essential to this approach is that the non-literal use of an adjective is signalled by a category error or by a value-expectation violation. A category error is found in a phrase like 'cold stare' where the 'stare' cannot actually be modified by a temperature estimation. The phrase 'cold steam' constitutes a value-expectation error: although the noun 'steam' has the property of temperature, the only permitted value is 'hot'. A structured connectionist spreading-activation model of semantic memory is used to represent the main assumption that the adjectives' literal usages can be exploited to find the metaphorical interpretation of an adjective–noun construct. Direct inferences are modelled by excitatory and inhibitory connections between noun units, property units and the property-value units. The primary mechanism for the interpretation of adjective–noun combinations is based on immediate inferences. One shortcoming of Hollbach-Weber's approach is that the noun is considered to be the only context for the metaphorical interpretation of an adjective. In the example 'cold shoulder' the system would interpret this phrase as metaphorically used. However, this phrase does not necessarily imply a metaphorical interpretation if we assume the context to be 'She touched the cold shoulder of the dead body'. Furthermore, this approach does not support a simple extension of the represented knowledge since the knowledge about metaphors must be encoded in advance as immediate inferences so that the system has only a limited generalization ability. Lastly, within an adjective–noun construct the noun may actually occur metaphorically, especially if more context than the pure phrase is available.

11.2.3 Overall analysis

We now summarize the main points of the existing symbolic and connectionist approaches to handling metaphor interpretation and point out some shortcomings.

- A distinction between literal and non-literal metaphorical language is unnatural (special-purpose routines for handling metaphorical language).

- Often only similarities between source and target objects are considered rather than the similarity of the metaphorical relationship.
- Knowledge about metaphors is hand-coded rather than learned.
- The symbolic representation of knowledge about metaphors does not support a graded plausible representation.
- There is no adequate consideration of explicit external context within metaphorical interpretation.

The approach we describe in the next section tries to overcome these shortcomings. It is based on learning metaphors in connectionist networks and using the generalization ability of the connectionist networks to interpret previously unknown metaphorical phrases based on plausible mappings.

11.3 A new connectionist model for metaphors

11.3.1 Motivation

A main aim of our approach is that the similarity between metaphorical concepts should be taken into account by a connectionist representation in order to support the interpretation of new mappings in a new context. For example, if we take the phrases 'thirsty airplane' and 'dead computer' both phrases are instances of the metaphor class 'artifact-as-living-thing'. The similarity of the mappings between the source concepts of both phrases and their target relationships should be considered in the interpretation of the phrases. To support this similarity between metaphorical mappings we use features like 'animate-property' and 'event-property' for adjectives and 'artifact' and 'moving-object' for nouns.

In general, metaphorical language is extremely sensitive to context and context directly influences the selection of a particular metaphorical mapping between the source objects and the target relationship. For example, in the phrase 'electronic thief' it depends on the context whether the interpretation of the metaphorical phrase will be a person or an artifact like a program. Our approach considers the influence of context by providing the means for directly modifying the mapping between the source concept (phrases as input to the system) and the target relationship (output of the system).

Connectionist networks can generate plausible interpretations by using a distributed representation of knowledge while static rules within symbolic systems fail to determine a plausible interpretation, especially in cases of incomplete and contradictory knowledge. This ability for plausible interpretation of connectionist networks supports the integration of metaphorical mappings. Furthermore using the learning capability of a connectionist network we do not have to encode explicitly the knowledge for metaphorical

interpretation since the connectionist network can learn from examples and generalize the interpretation to new, previously unknown, examples.

Table 11.1 shows some examples of phrases and the relationships describing the source objects and target relationships for literal and metaphorical phrases. For example, the source concept of the adjective–noun combination 'defective airplane' can be described as an 'artifact' having a specific 'artifact-property'. If the context for this phrase is not particularly set for a certain direction, this phrase is interpreted literally, which is expressed by a target relationship 'artifact-with-artifact-property'. On the other hand, the phrase 'thirsty airplane' is a metaphorical example. The source concept of this phrase can be described as an 'artifact' with an 'animate-property'. The target relationship is given by 'artifact-with-artifact-property' since 'thirsty' should be interpreted as 'consuming a lot of fuel'. That is, the relationship 'artifact-with-artifact-property' is the literal interpretation for 'defective airplane' but the metaphorical interpretation for 'thirsty airplane' is based on the underlying properties of the source objects.

Table 11.1 *Phrases, source objects and intended target relationship for literal and metaphorical interpretations of adjective–noun combinations*

Input phrase	Source objects	Target relationship
defective airplane	artifact, property	artifact-with-artifact-property
thirsty person	animate, property	animate-with-animate-property
catastrophic accident	event, property	event-with-event-property
thirsty airplane	artifact, property	artifact-with-artifact-property
dead computer	artifact, property	artifact-with-artifact-property

11.3.2 System architecture

An overview of our system is shown in Figure 11.1. Input to the system is a literally or metaphorically used phrase in a specific context, output of the system is a target relationship. For each phrase and its context the encoding component of the system produces an input pattern for the connectionist network. Depending on the phrase and the context, the network classifies the input pattern as belonging to a target relationship represented by a particular output unit. Finally, the output of the network is decoded in a target relationship for the input phrase and its context.

At present the system considers metaphors consisting of adjective–noun and noun–preposition–noun combinations, like 'magnetic person' or 'engine in person'. The training set for the connectionist network consists of

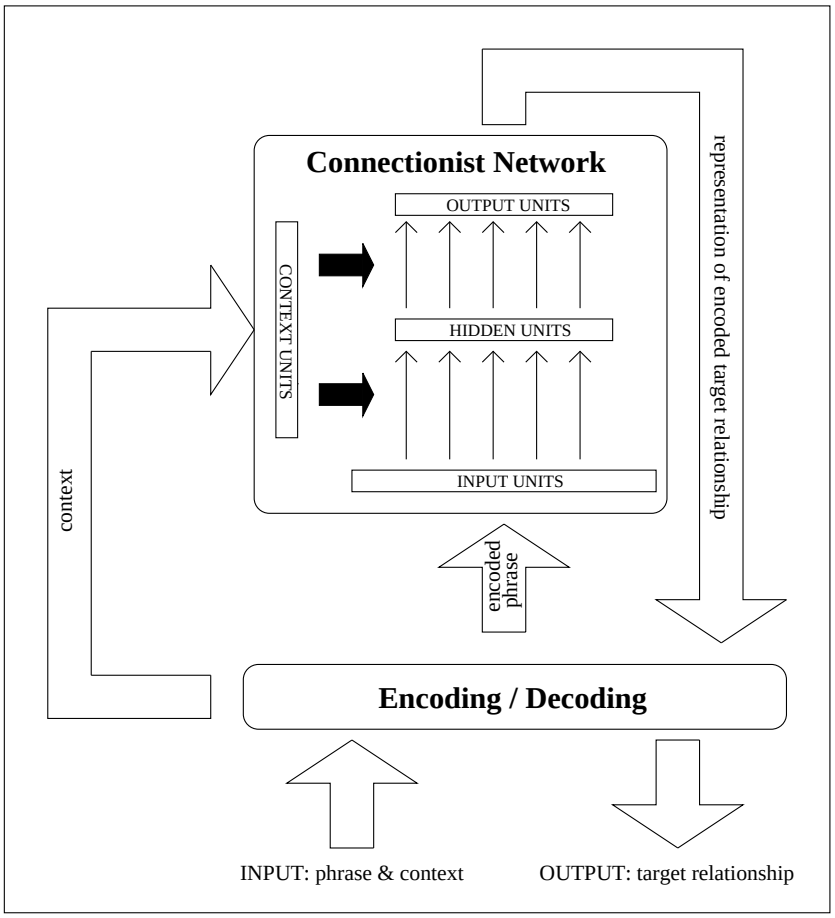

Figure 11.1 *System overview*

patterns, each describing the phrase, its context and the desired target relationship. The description of a phrase consists of seven features for the adjective and another seven features for the noun. The connectionist network classifies the input phrases into eight possible target relationships. Each of these is represented by a particular output unit.

Table 11.2 lists the features used to encode adjectives and nouns with a particular example: for instance, 'thirsty' is an 'animate-property' and a 'human-property'. Furthermore, the features 'moving-object', 'artifact', and 'location' describe the noun 'airplane'. The context for adjective–noun combinations is described by five features: 'animate', 'artifact', 'movement', 'location', and 'event'. All target relationships the input can potentially be mapped to are shown in Table 11.3. This limited set of features and relation-

Table 11.2 *Features and examples for adjectives and nouns in adjective–noun combinations*

Encoding position	Adjective features	Example: thirsty	Noun features	Example: airplane
1	animate-property	1	animate	0
2	human-property	1	human	0
3	moving-object-property	0	moving-object	1
4	artifact-property	0	artifact	1
5	location-property	0	location	1
6	event-property	0	event	0
7	abstract-object-property	0	abstract-object	0

ships was selected in a first attempt to represent metaphors in a learning connectionist system and so far they have been a sufficient description for the set of phrases examined.

The use of these features for encoding the phrase 'thirsty airplane' in context 'artifact' is shown in Table 11.4. The desired output of the system for this particular input is the target relationship 'artifact-with-artifact-property' indicated by the encoded output pattern '00010000' of the network as shown in the table.

Table 11.3 *Target relationships for adjective–noun combinations*

Decoding position	Target relationship
1	animate-with-animate-property
2	human-with-human-property
3	moving-object-with-moving-object-property
4	artifact-with-artifact-property
5	location-with-location-property
6	event-with-event-property
7	abstract-object-with-abstract-object-property
8	other-relationship

We pointed out earlier that similarity and context are important for detecting metaphorical relationships. Therefore we need an architecture which supports the influence of context by directly modifying the represented mappings. The architecture of our network consists of a function network

Table 11.4 *Examples of representations of metaphorical interpretations of adjective–noun combinations*

Input/ output	Specification	Example	Encoding/ decoding
input	adjective	thirsty	1100000
input	noun	airplane	0011100
input	context	artifact	01000
output	target relationship	artifact-with-artifact-property	0001000
input	adjective	electronic	0001000
input	noun	virus	1000000
input	context	artifact	01000
output	target relationship	artifact-with-artifact-property	0001000

which models the mapping between source objects and target relationships of metaphorical language and a context network which – depending on the context – modifies the weights of the function network. This architecture is inspired by an approach with a similar architecture to that of [Pollack 1987] although Pollack's network was developed for processing input sequentially by feeding the output of the function network back into the context network.

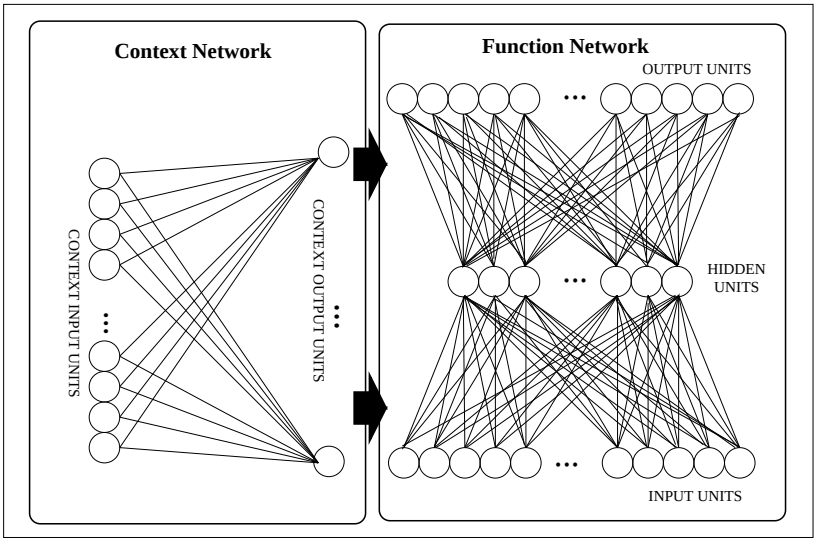

Figure 11.2 *Architecture of the connectionist network used*

Figure 11.2 shows the architecture of the connectionist network in more detail. A 3-layer feedforward network, the function network, is controlled by a 2-layer feedforward network, the context network. Each weight in the feedforward network is controlled by the context network. This is done by propagating the input of the context network to the context output units. For each weight in the function network we have a corresponding context output unit. These context units sum up the different weighted inputs they receive. The weights of the function network are then set to the activation of the corresponding context output units. After the weights of the function network have been set, the activation of the input units is propagated through the feedforward network.

We assume $\vec{x}$ to be the vector of i input activations $(x_1, \ldots, x_i)$, $\vec{h} = (h_1, \ldots, h_l)$ to be the vector of l activations of the hidden units, $\vec{z} = (z_1, \ldots, z_k)$ to be the vector of k output activations, and $\vec{y} = (y_1, \ldots, y_m)$ to represent the input to the context network. We describe the weights of connections between the input layer and the hidden layer of the function network by the matrix W_{li}. The matrix W_{kl} represents the weights between the units of the hidden layer and the output layer. The matrices C_{lim} and C_{klm} denote the weights of the context network, where C_{lim} consists of the weight from the context input units to those context output units that determine the weights of the connections between the input and hidden units of the function network. C_{klm} similarly describes the weights of the connections between the context input units and those units in the context output layers that determine the connections between hidden and output layer of the function network. Then, learning in this connectionist network takes place in the following four steps.

1. Propagate the activation of the context units and determine the weights of the function network by using

$$W_{li} = C_{lim} \cdot \vec{y}$$

$$W_{kl} = C_{klm} \cdot \vec{y}$$

2. Propagate the activation of the input units through the hidden units to the output units by using

$$\vec{h} = f(W_{li} \cdot \vec{x})$$

$$\vec{z} = f(W_{kl} \cdot \vec{h})$$

with

$$f(x) = \frac{1}{1 + \mathrm{e}^{-x}}$$

3. Determine and propagate the error backwards through the function network by using

$$\frac{\partial E}{\partial \vec{z}} = (\vec{z} - \vec{D})\vec{z}(1 - \vec{z}) \qquad \frac{\partial E}{\partial W_{kl}} = \frac{\partial E}{\partial \vec{z}} \times \vec{h}$$

$$\frac{\partial E}{\partial \vec{h}} = \frac{\partial E}{\partial \vec{z}} \cdot W_{kl}\vec{h}(1 - \vec{h}) \qquad \frac{\partial E}{\partial W_{li}} = \frac{\partial E}{\partial \vec{h}} \times \vec{x}$$

4. Propagate the error backwards through the context network using

$$\frac{\partial E}{\partial C_{lim}} = \frac{\partial E}{\partial W_{li}} \times \vec{y} \qquad \frac{\partial E}{\partial C_{klm}} = \frac{\partial E}{\partial W_{kl}} \times \vec{y}$$

During a training phase the function network is trained with the back-propagation learning rule [Rumelhart *et al.* 1986] to adjust the weights as described above. Although the weights of the function network will be set by the context network, the errors within the function network have to be computed since the errors in the function network are relevant to the computation of the weight adjustment in the context network. Once the network is trained, the weights of the context network are fixed and the weights of the function network change according to the output of the context network.

11.3.3 Results of learning and generalization

In our first set of experiments we selected 20 adjectives and 30 nouns which produced a total of 600 possible adjective–noun combinations. This selection was made after examination of metaphors in the corpus of the 'Treebank project' at the University of Pennsylvania. Several metaphors were taken from this corpus but not all of the generated adjective–noun combinations actually occurred in the corpus.

Since we wanted to examine the interpretation of the adjective–noun combinations in various contexts, we did not simply take the noun as context for the adjective interpretation [Hollbach-Weber 1989], but we examined adjective–noun combinations under the influence of five different classes of external context. Half of this set of 3000 context–adjective–noun combinations was used to train the connectionist network, which consisted of a function network with 14 input units (7 features each for noun and adjective), 10 hidden units and 8 output units (8 target relationships) as well as a context network with 5 input units (5 contexts).

To train the network we used 1500 patterns which were randomly chosen. Three runs with different patterns were made to make the results more independent of the chosen training patterns and the initial random-weight initialization of the network. Table 11.5 shows the results of the network for three different sets of randomly chosen training/test examples. Training was stopped when the average sum squared error for a pattern had been

reduced to 0.001. Using a learning rate of 0.2, this remaining error was reached after training the network for an average of 100 000 epochs where one epoch consists of the presentation of the whole training set.

Table 11.5 *Results of different runs of the network for adjective–noun combinations*

Context-function network	Error rates			
	Run1	Run2	Run3	Average
Training set	0.0%	0.0%	0.1%	0.0%
Test set	18.7%	11.2%	10.1%	13.3%

As there is only one unit in the target vector the activation of which is equal to 1 for a certain input, we compare its position with the position of the unit with the maximum activation value. If the positions agree we count the input as classified correctly, if not it is classified incorrectly. The network is able to associate the correct target relationship with all input phrases for all adjective–noun phrases in the training set. The results of the network on the test sets are considered satisfactory given that none of the phrase/context combinations of the test set appeared within the training set.

Table 11.6 shows some generated target relationships for different contexts and different adjective–noun phrases taken from the test set. For the first two examples the system proposes a *literal interpretation* of the phrases since the features of the source objects 'defective engine' and 'hungry person' agree with their respective context. Example 3 and 4 demonstrate that the same network can also represent *metaphorical interpretation* and can avoid a separation of metaphorical interpretation into specific additional representations. For these examples the metaphorical target relationship is chosen independently of the context, that is, for any of our different kinds of context the system selects the same target relationship for the phrase. On the other hand, the last two examples show the influence of context. The context 'artifact' increases the influence on the adjective in the input phrase 'dead engine' and the relationship 'artifact-with-artifact-property' is selected while the context 'animate' influences the features of the noun 'engine', resulting in choice of the 'animate-with-animate-property' target relationship.

As an alternative architecture we eliminated the context network and introduced the context directly into the function network using a 3-layer feedforward network with 19 input units (7 each for adjective and noun, 5 for the context), 15 hidden units and 8 output units (8 target relationships). This architecture resulted in similar results for the training set but the generalization results on the test set were worse than those of the context-

Table 11.6 *Generated target relationships for adjective–noun combinations*

	Phrase	Context	Target relationship
1	defective engine	artifact	artifact-with-artifact-property
2	hungry person	animate	human-with-human-property
3	dead conference	any	event-with-event-property
4	thirsty airplane	any	artifact-with-artifact-property
5	dead engine	artifact	artifact-with-artifact-property
6	dead engine	animate	animate-with-animate-property

function network. These results demonstrate that a separate context network has advantages with respect to the interpretation of adjective–noun combinations since there the context can directly modify the mapping.

In order to test the context-function network on a different task of metaphor interpretation we used a set of noun–preposition–noun combinations which were based on the metaphors from the Treebank corpus as well. We used 30 different nouns and the three prepositions 'at', 'in' and 'on' in two different kinds of context ('artificial' and 'natural').

Table 11.7 *Features for nouns in noun–preposition–noun combinations*

Encoding position	Noun features
1	animate
2	human
3	artifact
4	container
5	physical-object
6	spatial-object
7	institution
8	location
9	event
10	abstract-object

For the representation of the nouns we used features derived from the examinations of spatial relationships in locative expressions [Herskovits 1986] (see Table 11.7). The three prepositions were encoded by using a separate input unit for each of them. For example, the input phrase 'engine in person' was encoded as '001011000 010 110111000'. The output of the

network was represented with the target relationships as shown in Table 11.8.

Table 11.8 *Target relationships for noun–preposition–noun combinations*

Decoding position	Target relationship
1	spatial-object-at-location
2	spatial-object-at-event
3	person-at-artifact
4	physical-object-on-physical-object
5	spatial-object-in-container
6	person-in-institution
7	physical-object-in-event
8	event-in-event
9	location-in-event
10	animate-relationship
11	human-relationship
12	artifact-relationship
13	other-relationship

The context-function network for this task has 23 input units, 18 hidden units, 13 output units for the function network, and 2 input units for the context network. This network was trained with 2700 noun–preposition–noun combinations and tested with 2700 different combinations. The results of the training and testing are shown in Table 11.9. The trained network associates the correct target relationship with 98.3% of the input phrases from the training set. The trained networks are able to correctly associate the intended target relationship for 85.4% of the phrases from the test set.

Table 11.9 *Results of different runs of the network for noun–preposition–noun combinations*

Context-function network	Error rates			
	Run1	Run2	Run3	Average
Training set	2.0%	2.1%	1.1%	1.7%
Test set	16.5%	13.3%	14.2%	14.6%

In Table 11.10 examples of target relationships for different noun–preposition–noun phrases in context are shown. In example 1 the features of the source objects in 'person in accident' force the system to select the literal target relationship 'physical-object-in-event' and the context supports this mapping. In the second example the source objects in 'engine in airplane' were mapped onto the literal target relationship 'spatial-object-in-container'.[2] Examples 3 and 4 are cases in which a metaphorical target relationship is chosen independently from the context for 'brain in computer' and 'heart in car' respectively. For any kind of context the system selects an identical target relationship. In contrast, the metaphorical target relationships of the phrase 'engine in person' in examples 5 and 6 depend on the context. An 'animate' context forces the system to select 'animate-relationship' as target relationship while an 'artifact' context results in an 'artifact-relationship' target relationship for the phrase.

Table 11.10 *Generated target relationships for noun–preposition–noun combinations*

	Phrase	Context	Target relationship
1	person in accident	natural	physical-object-in-event
2	engine in airplane	artificial	spatial-object-in-container
3	brain in computer	any	artifact-relationship
4	heart in car	any	artifact-relationship
5	engine in person	natural	animate-relationship
6	engine in person	artificial	artifact-relationship

11.4 Learned internal representation for the adjective–noun phrases

After training had been completed we examined the internal representation of the phrases. Here we demonstrate representative results for the target relationships 'animate-with-animate-property', 'human-with-human-property' and 'artifact-with-artifact-property'. Figures 11.3 and 11.4 show the activation of the hidden units for 8 different examples of adjective–noun phrases. Each row contains the context, the input phrase, the kind of phrase (literal or metaphorical), the target relationship, and the activation of the 10 hidden units. In these figures we abbreviate the target relationship 'animate-with-animate-property' as 'ani-w-ani', 'human-with-human-property' as 'hum-w-hum', and 'artifact-with-artifact-property' as

[2] This is the desired target relationship according to the list of relationships shown in Table 11.7.

'art-w-art'. The activation of a hidden unit is within the interval [0,1]. Activation values between 0.0 and 0.5 are described by white squares of decreasing size, activation values between 0.5 and 1.0 by black squares of increasing size.

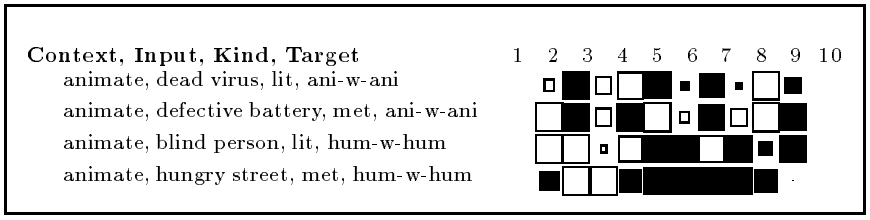

Figure 11.3 *Internal representation of specific target relationship*

The first two rows in Figure 11.3 show two examples of a literal and a metaphorical phrase with the target relationship 'animate-with-animate-property'. While a literal interpretation as 'animate-with-animate-property' is generated for the phrase 'dead virus', the desired metaphorical target relationship 'animate-with-animate-property' is generated for 'defective battery' since 'defective battery' occurred here as a metaphorical concept ('defective battery' is used in the sense of 'unhealthy heart', indicated by the 'animate context). Examining the internal representation of all 381 phrases with a desired 'animate-with-animate-property' target relationship we found that this target relationship correlated with a high value for hidden unit 2 and a low value for hidden unit 9.

The last two rows in Figure 11.3 show example phrases of a 'human-with-human-property' target relationship. The network generates a literal interpretation for 'blind person' but a metaphorical interpretation for 'hungry street' since 'hungry street' occurred as a metaphorical concept ('hungry street' is used for 'hungry people being in the street', indicated by the 'animate' context). Comparing the activation of the hidden units of the 205 phrases with the 'human-with-human-property' target relationship we found low values for hidden unit 2 and high values for hidden unit 8.

Similarly the internal representations for other target relationships showed certain combinations of significant hidden units for certain target relationships; for instance, all 360 phrases with a target relationship 'artifact-with-artifact-property' correlated with high values for hidden units 1 and 3.[3] These results show the ability of the network to handle literal as well as metaphorical phrases using a similar representation for both kinds of phrases. The correspondence of certain collections of hidden units with certain target relationships provides evidence that the network has learned an internal distributed representation for target relationships.

[3] Some instances are shown as examples 1, 2 and 4 in Figure 11.4.

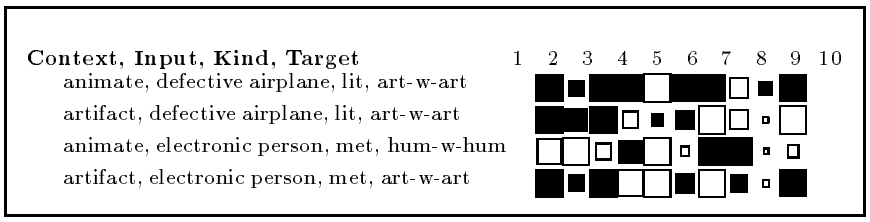

Figure 11.4 *Influence of the context 'animate' and 'artifact'*

While Figure 11.3 demonstrated the generation of literal and metaphorical interpretations within a constant context, we will now focus on the influence of the context on the interpretation of adjective–noun combinations. In Figure 11.4 we show the internal representation of the literal phrase 'defective airplane' and the metaphorical phrase 'electronic person' in the contexts 'animate' and 'artifact'. In the first two examples we see that the different context does not lead to different target relationships, since 'defective airplane' is associated with its literal 'artifact-with-artifact-property' relationship. On the other hand, the last two examples show that the phrase 'electronic person' is interpreted as a 'human-with-human-property' target relationship for the context 'animate' (that is, an interpretation as a person with a passion for electronics) and as an 'artifact-with-artifact-property' target relationship for the context 'artifact' (that is, an interpretation as a computer with the capabilities of a person). Furthermore, the internal representations of the two phrases 'defective airplane' and 'electronic person' are rather different although the input representation was the same and only the context was different. This indicates the influence of the context network on the function network. While the function network provides a basic functionality for the interpretation of literal and metaphorical target relationships, the context network controls the function network and induces different functionalities into the function network.

11.5 Discussion and conclusion

We have introduced a new connectionist approach to the interpretation of metaphorical language as an example for high-level natural-language processing. This approach has a number of properties which make it different from other previous approaches. The system *integrates literal and non-literal interpretation* in a single architecture. In contrast to many symbolic approaches there is no separation into a module for literal interpretation and an additional module for metaphor interpretation using special-purpose routines: the system *integrates the recognition and mapping problems* which are essential for the interpretation of metaphorical language.

A connectionist network is used to represent the metaphorical mapping between source and target concepts. Unlike Hollbach-Weber we use a connectionist network that determines a target relationship.

The results indicated that a connectionist representation can *learn plausible interpretations* in order to support the interpretation of metaphorical mappings and that the inherent generalization ability of connectionist networks can be used to perform well on previously unknown phrases. Furthermore we used an architecture that allows us to examine the influence of *explicit external context*. Therefore, phrases could be examined under the influence of specific context, which has not been done in most previous approaches. The context has an effect on the interpretation of a phrase by constraining the potential interpretations of a metaphor. It could be shown that an architecture with an explicit external context network in addition to the function network performed better than an architecture with the function network alone.

The association of a phrase with a potentially metaphorical target relationship is only one aspect of the complete analysis of a phrase. However, phrasal metaphors occur very frequently so that a metaphorical module serves an important function within a complete phrasal analysis. In contrast to [Martin 1988] who relies on a symbolic hand-coded in-depth understanding of complex metaphors we focused on a connectionist learned scanning understanding of metaphors with a relatively simple surface structure. However, our system is able to learn, represent and interpret a large set of such metaphors, tackling the problem of scaling up connectionist networks for difficult real-world problems. We conclude that such a scanning understanding has a lot of potential for dealing with a substantial number of real-world metaphors based on learning plausible similarity mappings, representing external context explicitly, and integrating literal and metaphorical processing.

11.6 References

Carbonell J.G.: Metaphor: inescapable phenomenon in natural-language comprehension, in Lehnert W.G., Ringle M.H. (eds.): *Strategies for Natural Language Processing*, Hillsdale, NJ: Erlbaum, pp. 415–434, 1982.

Cottrell G.W.: A model of lexical access of ambiguous words, in Small S.I., Cottrell G.W., Tanenhaus M.K. (eds.): *Lexical Ambiguity Resolution* , San Mateo, CA: Morgan Kaufmann, pp. 179–194, 1988.

Cottrell G.W., Small S.L.: A connectionist scheme for modelling word sense disambiguation, *Cognition and Brain Theory* **6**, 89–120, 1983.

Fass D.: Collative semantics: a semantic for natural language processing, Ph.D. Thesis, New Mexico State University, Las Cruces, New Mexico, CLR Report MCCS-88-118, 1988.

Gentner D.: Structure-mapping: a theoretical framework for analogy, *Cognitive Science* **7**, 155–170, 1983.

Gentner D., Falkenhainer B., Skorstad J.: Viewing metaphor as analogy: the good, the bad, and the ugly, in Wilks Y. (ed.): *Theoretical Issues in Natural Language Processing*, Hillsdale, NJ: Lawrence Erlbaum, pp. 171–177, 1989.

Herskovits A.: *Language and Spatial Cognition*, Cambridge University Press, 1986.

Hollbach-Weber S.: Figurative adjective–noun interpretation in a structured connectionist network, *Proceedings of the 11th Annual Conference of the Cognitive Science Society*, Hillsdale, NJ: Erlbaum, 1989.

Hollbach-Weber S.: A connectionist model of literal and figurative adjective–noun combinations, in Fass D., Hinkelman E., Martin J.H. (eds): *Proceedings of the IJCAI Workshop on Computational Approaches to Non-literal Language: Metaphor, Metonymy, Idiom, Speech Acts and Implicature*, Sydney, Australia, pp. 151–160, 1991.

Lakoff G.: *Women, Fire and Dangerous Things: What Categories Reveal about the Mind*, University of Chicago Press, 1987.

Lakoff G., Johnson M.: *Metaphors We Live By*, University of Chicago Press, 1980.

Martin J.H.: A computational theory of metaphor, University of California, Berkeley, Computer Science Department, Report No. UCB/CSD 88–465, 1988.

Pollack J.B.: On connectionist models of natural language processing, New Mexico State University, Computing Research Labroratory, MCCS-87-100, 1987.

Rumelhart D.E., Hinton G.E., Williams R.J.: Learning internal representations by error propagation, in Rumelhart D.E., McClelland J.L. (eds): *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol. 1, Cambridge, MA: MIT Press, pp. 318–364, 1986.

Small S., Rieger C.: Parsing and comprehending with word experts, in Lehnert W.G., Ringle M. (eds.): *Strategies for Natural Language Processing*, Hillsdale, NJ: Erlbaum, 1982.

Way E.C.: *Knowledge Representation and Metaphor*, Dordrecht: Kluwer Academic, 1991.

Wermter S.: Integration of semantic and syntactic constraints for structural noun phrase disambiguation, *Proceedings of the 11th International Joint Conference on Artificial Intelligence (IJCAI)*, Detroit, MI, pp. 1486–1491, 1989.

Wermter S.: A hybrid and connectionist architecture for a scanning understanding, in Neumann B. (ed.): *ECAI '92: Proceedings of the 10th European Conference on Artificial Intelligence*, 1992.

Wilks Y.: An intelligent analyzer and understander of English, *Communications of the ACM* **18(5)**, 264–274, 1975.

Zernik U.: Strategies in language acquisitions: learning phrases from examples in context, University of California, Los Angeles, Ph.D. thesis, Technical Report UCLA-AI-87-1, 1987.