# Applying Contextual Constraints in Sentence Comprehension

Mark F. St. John and James L. McClelland

Carnegie-Mellon University

The goal of our research has been to develop a model that converts a simple sentence into a conceptual representation of the event that the sentence describes: specifically, a model that converts the constituent phrases of a sentence into a representation of an event, that assigns a thematic role to each constituent (Fillmore, 1968), and that interprets ambiguous and vague words. In our model, the comprehension process is viewed as a form of constraint satisfaction. The surface features of a sentence, its particular words and their order and morphology provide a rich set of constraints on the sentence's meaning. These constraints vary in strength and compete or cooperate according to their strength to produce an interpretation of the sentence.

Determining the exact constraints, and their appropriate strengths, is difficult, but a connectionist learning procedure allows a model to learn them. The learning procedures take a statistical approach to the task. By comparing large numbers of sentences to the events they describe, the many-to-many mapping, between features of the sentences and events, emerges as statistical regularities.

Often a sentence will omit information about the event it describes. We wanted our model to infer this missing information: to represent the event as completely as possible. Sometimes, though, a sentence is compatible with more than one interpretation. In "The private shot the target," the instrument is sometimes a rifle, and sometimes a pistol. In such situations, a good long-term strategy is to represent each possibility according to its likelihood in the given context.

As each constituent of a sentence is processed, the context changes. The processor should update its inferences to reflect the changing context. It should also adjust its interpretation of previous material. The model should utilize information derived from the sentence immediately (Carpenter & Just, 1977; Marslen-Wilson & Tyler, 1980). In all, therefore, we have six goals for our model of sentence comprehension:

* to disambiguate ambiguous words
* to instantiate vague words
* to assign thematic roles
* to elaborate implied roles
* to learn to perform these tasks
* to immediately adjust its interpretation

## Overview of the Model

### Processing

A sentence is represented as a sequence of phrases and each is processed in turn. The information each phrase provides is immediately used to update a representation of the event. This representation is called the sentence gestalt because all of the information from the sentence is represented together within a single, distributed representation. The event described by a sentence is represented as a pattern of activity across the units of this representation.

To process the constituent phrases of a sentence, we adapted an architecture from Jordan (1986) that uses the output of previous processing as input on the next iteration. To process a phrase, the appropriate *phrase* units are activated and the *sentence gestalt* activations (initially zero) are copied over to the *previous sentence gestalt* units. Activation from these layers combine in a hidden layer and create a new pattern over the *sentence gestalt* units (see Figure 1). Each phrase of the sentence is processed in sequence.

Though other models have used a type of sentence gestalt to represent the meaning of a sentence (McClelland & Kawamoto, 1986; St. John & McClelland, 1987), ours is the first to make the gestalt a trainable layer of hidden units. The advantage is that the network can optimize its representation to include only information that is relevant to its task. Since a layer of hidden units cannot be trained directly, we invented a way of "decoding" the representation into an output layer. The output layer represents the event as a set of thematic role and filler pairs. For example, the event described by "The pitcher

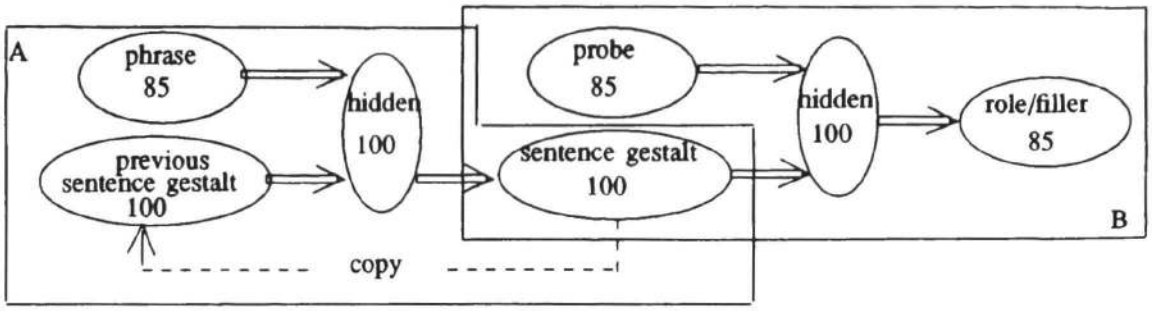


**Figure 1.** The architecture of the network. The boxes highlight the functional parts: Area A processes the phrases into the sentence gestalt, and Area B processes the sentence gestalt into the output representation.

threw the ball" would be represented as the set {(agent, pitcher/ball-player), (action, threw/toss), (patient, ball/sphere)}.

The output layer can represent one role/filler pair at a time. To decode a particular role/filler pair, the sentence gestalt is probed with half of the pair. Activation from the probe and the *sentence gestalt* combine in another hidden layer which then activates the entire pair in the output layer. The entire event can be decoded in this way by successively probing with each half of each pair.

When more than one object can plausibly fill a role, the model maximizes its long-term success by activating each filler to the degree it is likely. More formally, the activation of each filler should correspond to its conditional probability of occurring in the given context. The network should learn weights to produce these activations through training. To achieve this goal, we employed an error measure in the learning procedure, cross-entropy (Hinton, 1987), that converges on this goal:

$$C = - \sum_j [T_j \log_2 (A_j) + (1-T) \log_2 (1-A_j)]$$

where $T_j$ is the target activation and $A_j$ is the output activation of unit j. As with any connectionist learning procedure, the goal is to minimize the error measure or cost-function (cf. Hinton, 1987). When C is minimized across all the sentences of the training corpus, the activation of a particular output unit is equal to the conditional probability that whatever the unit represents is true given the current situation. Specifically, if each unit represented the occurrence of a particular filler in an event, that unit's activation would represent the conditional probability of that filler occurring given what was currently known about the event. In minimizing C, the network is searching for weights that allow it to match activations to conditional probabilities.

**Environment and training**

Training consists of trials in which the network is presented with a sentence and the event it describes. The network is trained to generate the event from the sentence as input. These sentence/event pairs were generated on-line for each training trial. Some pairs were more likely to be generated than others. Over training, these likelihood differences translated into differences in training frequency and created regularities.

To promote immediate processing a special training regime is used. After each phrase has been processed, the network is trained to predict the set of role/filler pairs of the entire sentence. From the first phrase of the sentence, then, the model is forced to try to predict the entire event. This training procedure forces the model to do as much immediate processing as possible. Consequently, as each new phrase is processed, the model's predictions of the event are refined to reflect the additional evidence it supplies.

**An illustration of processing**

As the network processes "The adult ate the steak," the *sentence gestalt* can be probed to see what it is representing after each phrase is processed (see Figure 2). After processing "the adult," the gestalt represents that the agent of the event is an adult, and it guesses weakly at a number of actions. Following "ate," the network encodes that information and expects the patient to be food. Once "the steak" is processed, the network represents steak as the patient and infers that knife is the instrument. It also is able to revise its representation of the agent. Because one person, the busdriver, is the most frequent steak-eater in the corpus, the network infers that the adult from the sentence is the busdriver. In this way, the model infers missing thematic roles and immediately adjusts previous results (in this case by instantiating the agent) as more information becomes available.

## Specifics of the Model

### Input representation

Each phrase was encoded by a single unit representing the noun or verb. No semantic information was provided. For prepositional phrases, the preposition was encoded by a second unit. The passive voice was encoded by a unit in the verb phrase representation. One unit stood for each of 13 verbs, 31 nouns, 4 prepositions, 3 adverbs, and 7 ambiguous words. Two of the ambiguous words had two verb meanings, three had two noun meanings, and two had a verb and a noun meaning. Six of the words were vague terms (e.g. someone, something, and food).

The relative location of each phrase within the sentence was also encoded in the input. It was coded by four units that represent location respective to the verb: pre-verbal, verbal, first-post-verbal, and n-post-verbal. The first-post-verbal unit was active for the phrase immediately following the verb, and the n-post-verbal unit was active for any phrase

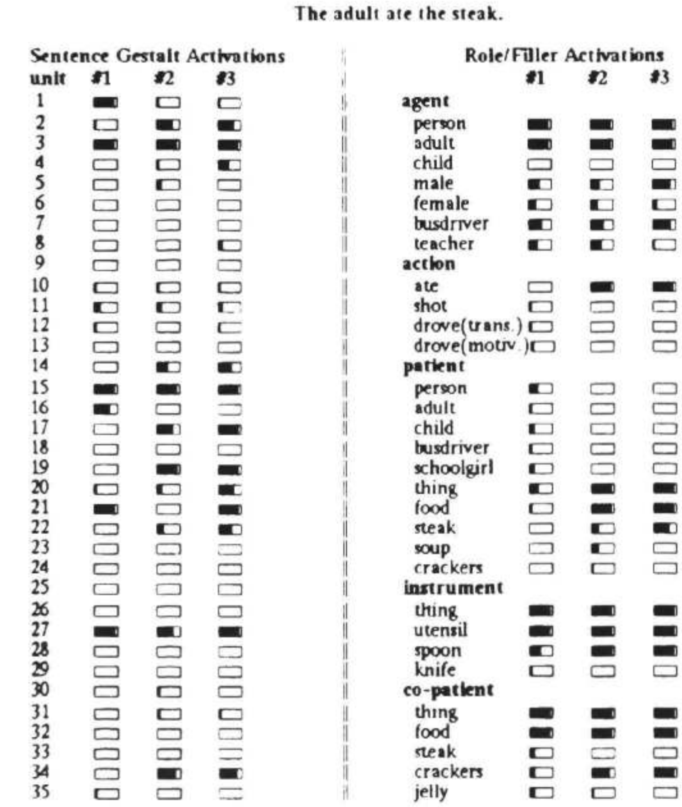


**Figure 2.** The evolution of the sentence gestalt during processing of a sentence. The # corresponds to the number of phrases that have been presented to the network at that point. #1 means the network has seen the first phrase; #2 means it has seen the first two phrases; etc. The activations (ranging between 0 and 1) of a sampling of units are graphed as the darkened area of each box. By probing the gestalt with the role half of each role/filler pair, the event represented by each gestalt can be observed. The role probe and the activation level of active filler units in the output layer are presented for each gestalt.

occurring after the first-post-verbal phrase. A number of phrases, therefore, could share the n-post-verbal position. For example, "The ball was hit by someone with the bat in the park," was encoded as the ordered set {(pre-verbal, ball), (verbal, passive-voice, hit), (first-post-verbal, by, someone), (n-post-verbal, with, bat), (n-post-verbal, in, park)}.

### Output representation

The output had one unit for each of 9 possible thematic roles (e.g. agent, action, patient, instrument) and one unit for each of 28 concepts, 14 actions, and 3 adverbs. Additionally, 13 "feature" units, like male, female, and adult, were included in the output to allow the demonstration of more subtle effects of constraints on interpretation. Any one role/filler pattern, then, consisted of two parts: for the role, one of the 9 role units was active, and for the filler, a unit representing the concept, action, or adverb was active. If relevant, any of the feature units or the passive voice unit were active.[1] This representation is not meant to be comprehensive. Instead, it is meant to be a convenient way to train and demonstrate the processing abilities of the network.[2]

### Training environment

While the sentences often include ambiguous or vague words, the events are always specific and complete: each thematic role related to an action is filled by some specific concept. Accordingly, each action occurs in a particular location, and actions requiring an instrument always have a specific instrument. The event would be generated by picking concepts to fill each thematic role related to the event according to preset probabilities. The probability of selecting each concept depended upon what else had been selected for that event. Conversely, the sentences often omit thematic roles (e.g. instrument and location) from mention and use vague words (e.g. someone and something) to refer to parts of an event.

The sentences had to be limited in complexity because of limitations in the event representation. Only one filler could be assigned to a role in a particular sentence. Also, all the roles are assumed to belong to the sentence as a whole. Therefore, no embedded clauses or phrases attached to single constituents were possible.

On each training trial, the error, in terms of cross-entropy, was propagated backward through the network (cf. Rumelhart, Hinton, & Williams, 1986). The weight changes from each trial were added together and used to update the weights every 60 trials. This consolidation of weight changes tends to smooth out the aberrations caused by the random generation of training examples.

## Performance

### Processing in general

The simulation was stopped and evaluated after 330,000 sentence trials. First, we will assess the model's ability to comprehend sentences generally. Then we will examine the model's ability to fulfill our specific processing goals. One hundred sentences were drawn randomly from the corpus. The probability of drawing a sentence was the same as during training. Consequently, frequently practiced sentences were more common among the 100 test sentences than infrequently practiced sentences. Thirteen of these sentences were completely ambiguous. For example, the sentence, "The adult drank the iced-tea," can be instantiated with either busdriver or teacher as the agent, but the sentence offers no clues that busdriver is the correct agent in this particular sentence/event pair in the test set.

---

[1] If the word related to the concept appeared in a prepositional phrase, such as "with the knife," the appropriate preposition unit was also activated.

[2] A second output layer was included in the simulations. This layer reproduced the input phrase that fit with the role/filler pair being probed. Consequently, the model was required to retain the specific words in the sentence as well as their meaning. Since this aspect of the processing does not fit into the context of the current discussion, these units are not discussed further.

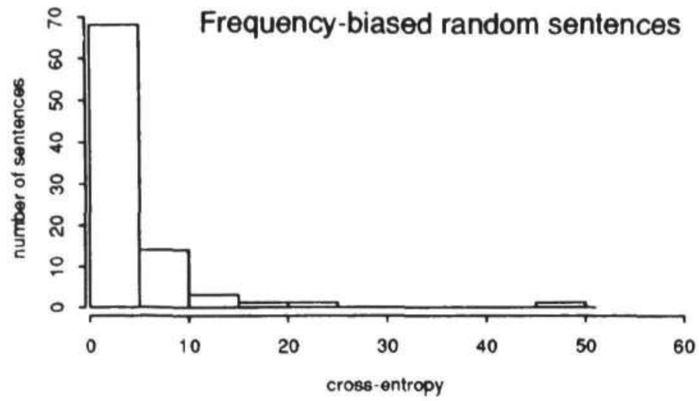


**Figure 3.** Histogram of the cross-entropy for frequency-biased random sentences. Sentences that were processed almost perfectly produced a small cross-entropy measure of between 0 and 10: only small errors occurred when an output unit should have been completely activate (with a value of 1), but only obtained an activation of .7 or .8, or when a unit should have had an activation of 0, but had an activation of .1 or .2. The incorrect activation of one role/filler pair produced a larger cross-entropy: between 15 and 20. For example, if teacher were supposed to be the agent, but the network activated busdriver, an error of about 15 would result.

Since the network cannot predict the event, these sentences were not tested. The remaining sentences were tested, and figure 3 presents a histogram of the results. Nearly every sentence was processed correctly.

**Performance on specific processes**

Our specific interest was to develop a processor that could correctly perform several important language comprehension processes. Five typical sentences were drawn from the corpus to test each processing task. The role assignment category was divided into groups according to the primary type of information used. Sentences in the active-voice semantic group contain semantic information relevant to assigning roles. In the example in Table 1, the network assigns schoolgirl to the role of agent. Of the concepts referred to in the sentence, only the schoolgirl has features which match an agent of stirring. Similarly, semantic knowledge constraints kool-aid to be the patient. Sentences in the passive-voice semantic category work similarly. The model processed each test sentence correctly.

To process sentences in the active and passive word order categories, however, the network cannot rely entirely on semantic constraints to assign thematic roles. In the sentence, "The busdriver was kissed by the teacher," the busdriver is as likely to be the

**Processing tasks**

| **Category** | **Example** |
|---|---|
| Role assignment | |
| Active semantic | The schoolgirl stirred the kool-aid with a spoon. |
| Passive semantic | The ball was hit by the pitcher. |
| Active word order | The busdriver gave the rose to the teacher. |
| Passive word order | The busdriver was kissed by the teacher. |
| Word ambiguity | The pitcher hit the bat with the bat. |
| Concept instantiation | The teacher kissed someone. |
| Role elaboration | The teacher ate the soup (with a spoon). |

**Table 1.** The four categories of processing tasks and an example sentence of each. Role assignment was tested under four conditions to assess the use of both semantic-and word order information. The remaining three categories involve inferences about the content of a sentence. The parentheses in the role elaboration example denote a role that is not presented in the sentence, and must be inferred from the context.

agent as the patient. Only the relative location information, in conjunction with the passive cues, can cue the correct role assignments. In active sentences, the pre-verbal phrase is the agent and the post-verbal phrase is the patient. In passive sentences, on the other hand, the pre-verbal phrase is the patient and the post-verbal phrase is the agent. The model processed each test sentence correctly.

The remaining three categories involve the use of context to further specify the concepts referred to in a sentence. Sentences in the word ambiguity category contain ambiguous words. While the word itself cues two different interpretations, the context fits only one. In "The pitcher hit the bat with the bat," pitcher cues both container and ball player. The context cues both ball player and busdriver because the model has seen sentences involving both people hitting bats. All the constraints supporting ball player combine, and together they win the competition for the interpretation of the sentence. Even when several words of a sentence are ambiguous, the event which they support in common dominates the disparate events that they support individually. Consequently, the final interpretation of each word fits together into a globally consistent event. For each test sentence, even when several words were ambiguous, the model performed correctly.

Concept instantiation works similarly. Though the word cues a number of more specific concepts, only one fits the context. Again, the constraints from the word and from the context combine to produce a unique, specific interpretation of the term. Depending upon the sentence, however, the context may only partially constrain the interpretation. Such is the case in "The teacher kissed someone." "Someone" could refer to any of the four people found in the corpus. Since, in the network's experience, females only kiss males, the context constrains the interpretation of "someone" to be either the busdriver or the pitcher, but no further. Consequently, the model activates the male and person features of the patient while leaving the units representing busdriver and pitcher partially and equally active. In general, the model is capable of inferring as much information as the evidence permits: the more evidence, the more specific the inference.

Finally, sentences in the role elaboration category test the model's ability to infer thematic roles not mentioned in the input sentence. For example, in "The teacher ate the soup," no instrument is mentioned, yet a spoon can be inferred. Here, the context alone provides the constraints for making the inference. Extra roles that are very likely are inferred strongly. When the roles are less likely, or when more than one concept can fill a role, the concepts are only weak inferred. Again, the model processed each test sentence correctly.

## Conclusions

Parallel Distributed Processing models have a number of qualities that make them useful as models of language comprehension. First, they allow the simultaneous processing of many constraints. Each bit of relevant information can be applied to a computation. This feature allows far greater interaction between constraints in computing an interpretation of a sentence. As Marcus (1980) points out, the competition and cooperation among constraints of varying strength is an essential aspect of comprehension. Second, PDP models allow the strength of constraints to vary on a continuum rather than discretely. Information may be more or less reliable, and correlations may be more or less strong. The activation level of a unit allows the quality of the information to be represented explicitly and combined effectively. This feature allows *quantitative* interaction among constraints. Third, PDP models naturally perform pattern completion. Given some input, the models add default information to produce an elaborated representation. Importantly, this default information is tailored to the specific input presented. This feature allows context specific inferences about the input to be computed.

Because PDP models learn, useful and complex constraints develop of their own accord. This learning process is slow, requiring a great deal of practice. While comprehension involving strong and regular constraints are learned relatively rapidly, irregular and complex constraints are only learned very slowly. Early on, therefore, the network begins to perform passably, and it slowly improves to correctly process more sentences.

The model handles ambiguity robustly by processing the input in terms of constraints on a conceptual representation of an event. Within this framework, word disambiguation and concept instantiation are similar processes. Both ambiguous words and vague terms provide conflicting constraints on their interpretation. The constraints from the word itself and additional constraints from its context provide evidence, respectively, for the general and specific features of the concept referred to. Role elaboration is similar, but at the level of interpreting the sentence as a whole. The features of the sentence provide constraints on the unspecified features of the event, such as implicit thematic roles like location, instrument, and manner. When more than one role/filler is likely, each is represented according to its conditional probability.

The interpretation of a sentence evolves as each word is presented to the network. The model sequentially processes each word and immediately adjusts its interpretation of old information and creates expectations of additional information as it attempts to represent the entire conceptual event.

The model cannot, however, represent sentences with embedded clauses. Extending the model to represent more complex sentences is an important goal. Meanwhile, we have learned a great deal about viewing sentence comprehension as a process of weak constraint satisfaction.

## References

Carpenter, P. A. & Just, M. A. (1977). Reading comprehension as the eyes see it. In M. A. Just & P. A. Carpenter (Eds.), *Cognitive processes in comprehension.* Hillsdale, NJ: Erlbaum.

Fillmore, C. J. (1968). The case for case. In E. Bach & R. T. Harms (Eds.), *Universals in linguistic theory.* New York: Holt, Rinehart, & Winston.

Hinton, G. E. (1987). Connectionist learning procedures. Tech report #CMU-CS-87-115.

Jordan, M. I. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. Paper presented to the 8th Annual Conference of the Cognitive Science Society. Amherst, MA.

MacWhinney, B. (1987). Competition. In B. MacWhinney (Ed.), *Mechanisms of language acquisition: The 20th annual Carnegie symposium on cognition.* Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Marcus, M. P. (1980). *A theory of syntactic recognition for natural language.* Cambridge, MA: MIT Press.

Marslen-Wilson, W. & Tyler, L. K. (1980). The temporal structure of spoken language understanding. *Cognition, 8,* 1-71.

McClelland, J. L. & Kawamoto, A. H. (1986). Mechanisms of sentence processing: Assigning roles to constituents. In J. L. McClelland, D. E. Rumelhart, and the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition, Volume 2.* Cambridge, MA.: MIT Press.

Rumelhart, D. E., Hinton, G. E., Williams, R. J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, and the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition, Volume 1.* Cambridge, MA.: MIT Press.

St. John, M. F. & McClelland, J. L. (1987). Reconstructive memory for sentences: A PDP approach. Paper presented to the Ohio University Inference Conference, *Proceedings Inference: OUIC 86.* University of Ohio, Athens, OH.