Title
Using Context to Interpret Indifrect Requests in a Connectionist Model of NLU

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 15(0)

Author
Schulenburg, David A.

Publication Date
1993

Peer reviewed

Using Context to Interpret Indirect Requests in a Connectionist Model of NLU

David A. Schulenburg
Department of Information & Computer Science
University of California
Irvine, CA 92717
schulenb@ics.uci.edu

Abstract

The role of context in natural language understanding (NLU) is generally accepted as being important to the task of ascertaining "correct meaning." This is particularly true during interpretation of otherwise ambiguous language constructs, such as lexical ambiguity resolution, metaphor understanding, and indirect speech act interpretation. This paper presents a feedforward connectionist model called SAIL2 which utilizes the context obtained from the processing of previously seen text to help resolve the ambiguity inherent in indirect speech acts, specifically, in indirect requests.

Introduction

The role of context in natural language understanding (NLU) is generally accepted as being important to the task of ascertaining "correct meaning." As Jarvella and Klein (1982) put it: "It is not difficult to understand, and quite natural, that language in its ordinary use is dependent on context." This is particularly true during interpretation of otherwise ambiguous language constructs, such as lexical ambiguity resolution (Small, Cottrell, and Tanenhaus, 1988), metaphor understanding (Shinjo and Myers, 1987), and indirect speech act interpretation (Gibbs, 1981). This paper presents a feedforward connectionist model called SAIL2 which is based on the simple recurrent network (Elman, 1990) and the story gestalt (St. John, 1992). SAIL2 utilizes the context obtained from the processing of previously seen text to help resolve the ambiguity inherent in indirect speech acts, specifically, in indirect requests. The context is used during the processing of the words of individual sentences in order to develop their "correct" pragmatic interpretation.

Consider the following two pieces of text:

(a) The survivors of the airplane crash gathered around the captain. He was taking inventory of their goods. He asked a passenger, "Do you have any matches?"

(b) The survivors of the airplane crash gathered around the captain. He took out his cigarettes but couldn't find his lighter. He asked a passenger, "Do you have any matches?"

In text (a), the question Do you have any matches? is intended to be interpreted literally; the captain wishes to know whether or not the passenger has any matches. However, in text (b) the question is intended as an indirect request for the passenger to give the captain a match to light his cigarette. Clearly, the context in which the question is put forth determines (in part) the interpretation that should be given the indirect request. This context should be available during sentence processing to help to derive the intended interpretation.

Feedforward connectionist models of NLU traditionally incorporate context into their understanding process by use of the simple recurrent network topology (Elman, 1990). However, this is not done on a "global" level. Rather, the recurrent relation is typically used at the "intra-sentence" processing level when processing single sentences or at the "inter-sentence" level when processing multiple sentence texts. That is to say, in single sentence processing, as the words of the sentence are presented to the network sequentially, the context represented by the sentence processing to that point is introduced as part of the input layer via a (simple) recurrent relation (Miikkulainen & Dyer, 1991; Schulenburg, 1992). In systems which process multiple sentence texts, the current context (the meaning of the text processed so far), as captured by the unit activation of some "internal" (hidden or output) layer of the network, is introduced as part of the input along with the meaning representation of the current sentence in order to produce an updated context (St. John, 1992). These systems do not combine the different contexts (those derived during intra-sentence

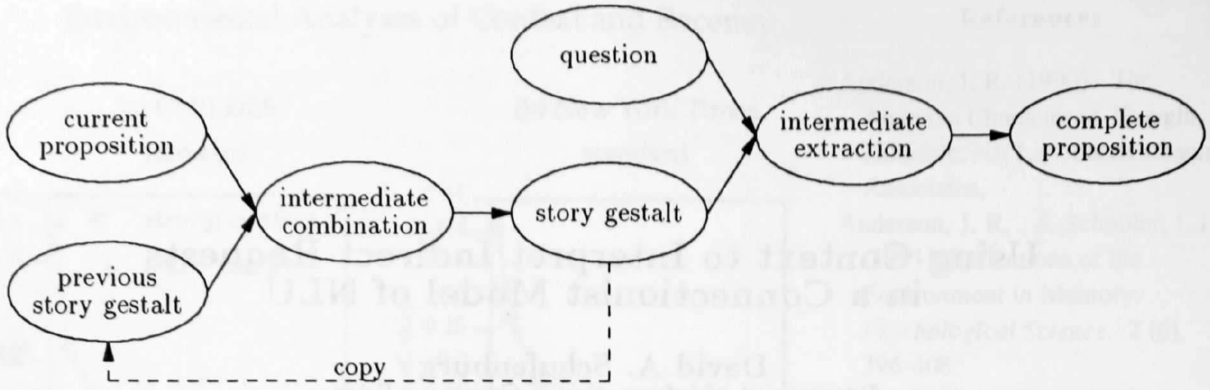

Figure 1: The Story Gestalt Network Topology.[1]

and inter-sentence processing) to form part of the input during intra-sentence processing; i.e., the intra-sentence processor does not benefit from the context produced during inter-sentence processing.

The Story Gestalt

St. John (1992) presents a connectionist model of text comprehension which processes propositions (the "meaning" of individual sentences) at the inter-sentence processing level into a learned representation (the *story gestalt*, capturing the "meaning" of the entire text). Figure 1 shows the general network topology used by the story gestalt model. The model uses a simple recurrent network to encode the sentence propositions into the story gestalt sequentially; as each new *current proposition* (i.e., each new sentence) is presented, it is combined with the *previous story gestalt*, fed through the *intermediate combination* hidden layer, to produce the current *story gestalt*. A "question/answering" regimen is utilized to train the "hidden" *story gestalt* layer. A *question* (represented by the single predicate of a proposition) is presented along with the current *story gestalt*; this is then fed through the *intermediate extraction* hidden layer to form the *complete proposition*, which is an exact copy of some proposition previously presented to the model and is represented by the predicate forming the *question*. The error is computed and backpropagated through the network all the way back to the current proposition and previous story gestalt input layers. The model has the ability to train the story gestalt as a "hidden" layer thus creating a representation for the entire text. This is what I call the inter-sentence processing level. (St. John's story gestalt model does not do intra-sentence processing; it uses pre-encoded *propositions* as sentence input.)

[1]From St. John (1992).

SAIL2

SAIL2 is a feedforward connectionist natural language model which uses a simple recurrent network topology to process sentences of a text sequentially, building a representation of the text by use of the story gestalt. Figure 2 shows the network topology used by SAIL2.[2]

Input to SAIL2 is a set of English sentences comprising an integrated text. The words of each sentence are processed sequentially by the intra-sentence processor. (See Schulenburg (1992) for an explanation of the basic sentence processing at the intra-sentence level used by SAIL2.) As the processing for each sentence is completed, its meaning representation is incorporated into the story gestalt in the same manner as that used by St. John, with one exception; the representation of the question presented for training the gestalt is different. In SAIL2, the *question* input layer is an entire representation for a question instead of the single predicate used by St. John; i.e., it is what an intra-sentence processor might output as the representation for a question. E.g. a question might be **Who went to the store?** which could have the following case/role assignments:

```
[(act go) (agent *?*) (to store)]
```

where `*?*` represents the unknown filler. The answer given in the *complete proposition* output layer (used to backpropagate the error), of course, would include the unknown filler. This approach does not limit the "questions" to one question for each predicate type per text.

[2]Note that the network topology implementing the story gestalt as seen in Figure 1 is wholly contained within the network topology used by SAIL2 as inter-sentence processing.

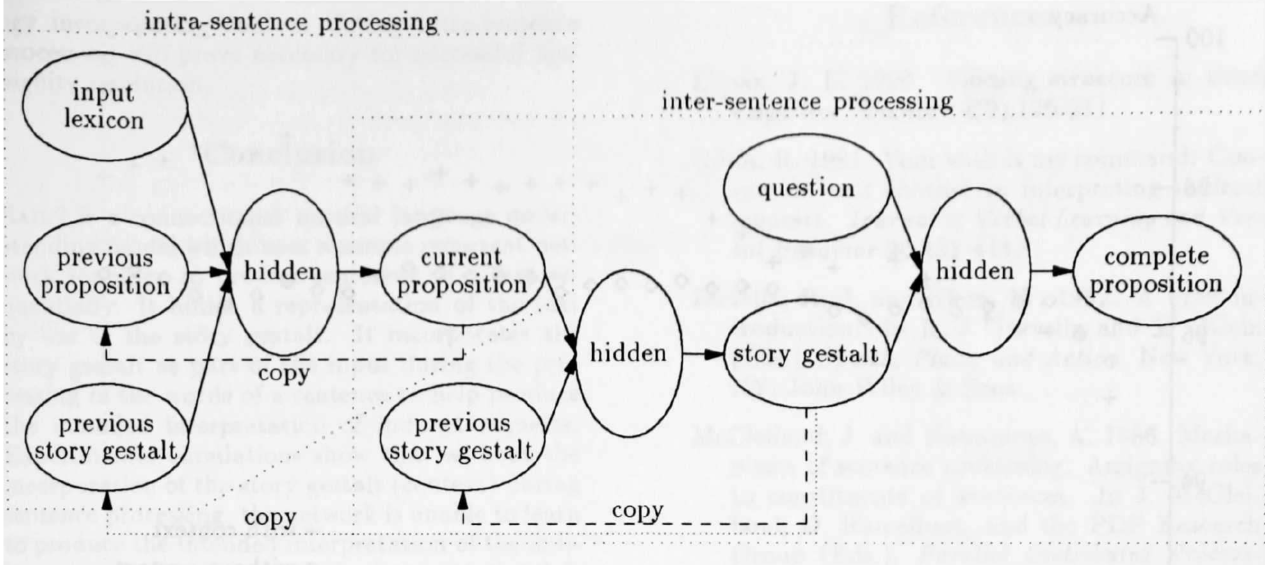

Figure 2: SAIL2 Network Topology.

During intra-sentence processing, the current context of the entire input text as captured by the *story gestalt* forms part of the input, along with the words of the sentence and the current sentence context. This is the main feature which distinguishes SAIL2 from other connectionist NLU models. (See the recurrent relation represented by the dotted line going from the *story gestalt* to the *previous story gestalt* forming part of the sentence input layer in Figure 2.) This added input allows the intra-sentence processor to use the context set up by the previous text to help disambiguate any potentially ambiguous (semantic) language constructs existing in the current sentence.

A series of simulations were run using SAIL2's network topology both with and without the story gestalt context aiding the intra-sentence processing to test the hypothesis that, with the context, ambiguous language constructs (in this case, indirect requests having both literal and indirect interpretations) will be properly disambiguated and that without the context, no disambiguation occurs. The data used in these simulations are three to four sentence texts (see the Appendix). Each text sets up a context for the final sentence (an indirect request) of the text which determines whether that final sentence has a literal or an indirect interpretation. The data set was divided into a training set and a testing set. Each text was generated with a 0.9 probability of being included in the training set; if a text was not in the training set, it was in the testing set. For the simulations reported here, there were 1031 texts in the training set and 121 texts in the testing set. Each hidden layer of the network had 25 units; the *story gestalt* had 15 units. The *input lexicon* layer had 55 units; the *current proposition* layer had 86 units. There were 108 units in the *question* layer and 103 units in the *complete proposition* layer. Initial weights on the connections were randomly generated in the interval $(-0.3, 0.3)$. Backpropagation of the simple recurrent network was used as the training algorithm.

The graph depicted in Figure 3 shows the overall percentage of correct[3] units in the *current proposition* output layer for the final sentences (the indirect requests) in the testing set of texts for two simulations, one simulation with the story gestalt context aiding the intra-sentence processing and one without. At first glance, it appears that there is not much difference between the accuracy shown by the two simulations; this conclusion, however, is misleading. After 600 training epochs (where the network incorporating the story gestalt context begins to show signs of overtraining), there is an approximately two percent difference between the accuracy for the two simulations. For these simulations, two of the 86 units in the *proposition* layer (or 2.3%) distinguish between the representations[4] for the direct and indirect interpretations of the final (indirect request) sentence of the text. Inspection of the activation of the "ambiguous" units shows that they are correct for the simulation incorporating the story gestalt into intra-sentence processing and that they do not reach their target values in the simulation without the aid of the story gestalt. Thus, the difference

[3] Output units are "correct" if their activation is within 0.1 of their target value.

[4] The representation used by SAIL2 for the *current proposition* is localist implying that few of these units are actually ambiguous.

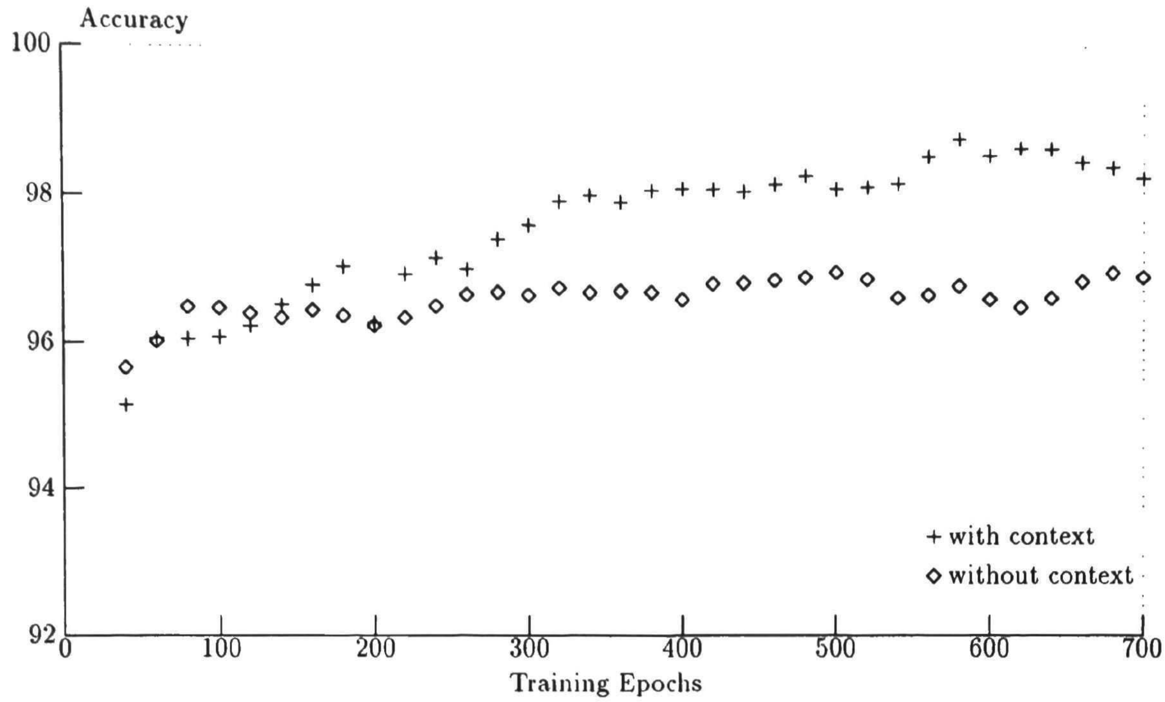

Figure 3: Percentage of Correct Units for Final Sentences in Text.

n accuracy can be attributed directly to the dif- 'erence between the two simulations, namely, that)ne simulation incorporates the story gestalt con- .ext during intra-sentence processing and one does 1ot. Also, the accuracy of the network topology vithout the story gestalt context asymptotes ear- ier and to a lower percentage; this is to be ex-)ected because that network topology cannot cor- 'ectly compute the activation for the ambiguous Inits. Thus, the story gestalt context is necessary luring intra-sentence processing to disambiguate ndirect requests.

Related Work

This section briefly reviews two other connection- st models which process multi-sentence texts. In the first, Miikkulainen and Dyer (1991) discuss DISPAR, their system which processes script-based stories into a slot-filler representation capturing the causal chain and role bindings of the story. Processing stories in DISPAR is accomplished via two "parsers," a sentence parser (intra-sentence processing) and a story parser (inter-sentence pro- cessing), each of which is implemented as a sim- ple recurrent network. These parsers act indepen- dently; the output of the sentence parser is fed as input to the story parser. Unlike in SAIL2, there is no story context given to the sentence parser. Because of this lack, DISPAR would not be able to parse input which required context for purposes of disambiguation. Furthermore, the representation used by DISPAR limits its usefullness to script-based stories. St. John (1992) presents the STORY GESTALT, an idea which SAIL2 heavily draws upon. Because the STORY GESTALT builds its own representation of the text, it is not limited to the script-based stories of DISPAR. As presented, however, there are some differences between St. John's work and that presented here. In the STORY GESTALT, the sentences themselves have been preprocessed. St. John rightfully argues that a preprocessor, such as the SENTENCE GESTALT (St. John & McClelland, 1990) could be added giving a topology similar to that of Miikkulainen and Dyer; however, the STORY GESTALT would still suffer from the inability to process individual sentences requiring context for purposes of disambiguation.

Future Work

SAIL2 is an ongoing experiment in connectionist NLU. Specifically, SAIL2 is designed to explore the importance of context during processing of ambiguous language constructs such as indirect requests. In the future, testing with SAIL2 will be expanded to other types of ambiguity, such as lexical ambiguity, metaphor, and other types of indirect speech acts. The hypothesis is that SAIL2's topology incorporating context during intra-sentence processing will prove necessary for successful ambiguity resolution.

Conclusion

SAIL2 is a connectionist natural language understanding model which uses a simple recurrent network topology to process sentences of a text sequentially. It builds a representation of the text by use of the story gestalt. It incorporates the story gestalt as part of the input during the processing of the words of a sentence to help produce the intended interpretation of indirect requests. Experimental simulations show that without the incorporation of the story gestalt (context) during sentence processing, the network is unable to learn to produce the intended interpretation of the indirect requests; furthermore, the interpretation left by the network in this case is ambiguous. Other existing connectionist NLU systems which process multiple-sentence text do not exploit the (story gestalt) context in this way and therefore do not have the ability to disambiguate potentially ambiguous input.

Appendix

The testing and training texts were created using a text template, similar to the way used by McClelland and Kawamoto (1986). Given slots of the template were filled with various words yielding specific texts. Two such templates are given below; the first indicates a literal interpretation of the question Do you have a(n) *object*? while the second indicates the indirect interpretation. The slots to be filled are indicated by the italicized words. All punctuation was included as separate input nodes to the intra-sentence processor.

- The survivors of the *disaster* gathered around the *leader*. The *leader* inventoried the assets. The *leader* asked the *person*, "Do you have a(n) *object*?"
- The *person* went to *class*. The *person* needed a(n) *object2* for the *event*. The *person* did not have a(n) *object2*. The *person* asked the T/A, "Do you have a(n) *object2*?"

Example fillers included:

- *leader:* captain doctor fire-fighter
- *person:* man woman boy girl
- *object2:* pencil book notebook
- *event:* exam quiz test experiment

References

Elman, J. L. 1990. Finding structure in time. *Cognitive Science* 14(2):179-211.

Gibbs, R. 1981. Your wish is my command: Convention and context in interpreting indirect requests. *Journal of Verbal Learning and Verbal Behavior* 20:431-444.

Jarvella, R. J. and Klein, W. 1982. A brief introduction. In R. J. Jarvella and W. Klein (Eds.), *Speech, Place, and Action*. New York, NY: John Wiley & Sons.

McClelland, J. and Kawamoto, A. 1986. Mechanisms of sentence processing: Assigning roles to constituents of sentences. In J. McClelland, D. Rumelhart, and the PDP Research Group (Eds.), *Parallel Distributed Processing: Explorations in the Microstructures of Cognition, Vol 2: Psychological and Biological Models*. Cambridge, MA: MIT Press.

Miikkulainen, R. and Dyer, M. G. 1991. Natural language processing with modular PDP networks and distributed lexicon. *Cognitive Science* 15:343-400.

Schulenburg, D. A. 1992. Sentence processing with realistic feedback. In *Proceedings of the International Joint Conference on Neural Networks '92*, IV-661-666. Piscataway, NJ: IEEE.

Shinjo, M. and Myers, J. L. 1987. The role of context in metaphor comprehension. *Journal of Memory and Language* 26:226-241.

Small, S. L., Cottrell, G. W., and Tanenhaus, M. K. 1988. *Lexical Ambiguity Resolution*. San Mateo, CA: Morgan Kaufmann Publishers, Inc.

St. John, M. F. 1992. The story gestalt: A model of knowledge-intensive processes in text comprehension. *Cognitive Science* 16:271-306.

St. John, M. F. and McClelland, J. L. 1990. Learning and applying contextual constraints in sentence comprehension. *Artificial Intelligence* 46:217-257.