To appear in *Proceedings of the 25th Annual Conference of the Cognitive Science Society (COGSCI-03, Boston, Massachusetts)*, 2003.

Incremental Nonmonotonic Parsing through Semantic Self-Organization

Marshall R. Mayberry, III (martym@cs.utexas.edu)
Risto Miikkulainen (risto@cs.utexas.edu)
Department of Computer Sciences
The University of Texas, Austin, TX 78712

Abstract

Subsymbolic systems have been successfully used to model several aspects of human language processing. Subsymbolic parsers are appealing because they allow combining syntactic, semantic, and thematic constraints in sentence interpretation and revising that interpretation as each word is read in. These parsers are also cognitively plausible: processing is robust and multiple interpretations are simultaneously activated when the input is ambiguous. Yet, it has been very difficult to scale them up to realistic language. They have limited memory capacity, training takes a long time, and it is difficult to represent linguistic structure. In this study, we propose to scale up the subsymbolic approach by utilizing semantic self-organization. The resulting architecture, INSOMNET, was trained on semantic representations of the newly-released LINGO Redwoods HPSG Treebank of annotated sentences from the VerbMobil project. The results show that INSOMNET is able to accurately represent the semantic dependencies while demonstrating expectations and defaults, coactivation of multiple interpretations, and robust parsing of noisy input.

Introduction

A number of researchers utilize neural network (i.e., subsymbolic) models to gain insight into human language processing. Such systems develop distributed representations automatically, giving rise to a variety of interesting cognitive phenomena. For example, neural networks have been used to model how syntactic, semantic, and thematic constraints are seamlessly integrated to interpret linguistic data, lexical errors resulting from memory interference and overloading, aphasic and dyslexic impairments resulting from physical damage, biases, defaults and expectations that emerge from training history, as well as robust and graceful degradation with noisy and incomplete or conflicting input (Allen and Seidenberg, 1999; McClelland and Kawamoto, 1986; Miikkulainen, 1997, 1993; Plaut and Shallice, 1993; St. John and McClelland, 1990).

Yet, despite their many attractive characteristics, neural networks have proven very difficult to scale up to parsing realistic language. Training takes a long time, fixed-size vectors make learning long-distance dependencies difficult, and the linguistic formalism used can impose architectural constraints, such as binary parse trees that are very deep and force more information to be compressed in higher nodes, thereby making the sentence constituents harder to recover. Progress has been made by introducing a number of shortcuts such as concentrating on small artificial corpora with straightforward linguistic characteristics (Berg, 1992; Ho and Chan, 2001; Sharkey and Sharkey, 1992), building in crucial linguistic heuristics such as Minimal Attachment and Right Association (Lane and Henderson, 2001; Mayberry and Miikkulainen, 1999), or foregoing parse structures altogether in order to concentrate on more tractable subproblems such as clause identification (Hammerton, 2001) and grammaticality judgements (Lawrence et al., 2000; Allen and Seidenberg, 1999; Christiansen and Chater, 1999). However, a promising new approach scales up a detailed artificial grammar to reflect frequency of structures from the Penn Treebank (Marcus et al., 1993) to account for a wide variety of psycholinguistic phenomena (Rohde, 2002).

Why is subsymbolic parsing a desirable goal? The main promise for both cognitive modeling and engineering is that it accurately accounts for the holistic nature and nonmonotonicity of natural language processing. Over the course of the parse, the network maintains a holistic parse representation at the output. Words processed later in a sentence can change the developing representation so that the network can recover from incorrect earlier decisions. This way, the network can more effectively resolve lexical ambiguities, attachments, and anaphoric references during the course of parsing. Indeed, multiple interpretations are maintained in parallel until disambiguating information is encountered in the input stream (cf. Onifer and Swinney, 1981; MacDonald et al., 1992; MacDonald, 1993). This is evidently how humans process natural language, what good parsers should do, and what subsymbolic parsers promise to deliver.

The purpose of the present study is to show that deep semantic parsing of sentences from real-world dialogues is possible using neural networks: a subsymbolic system can be trained to read a sentence with complex grammatical structure into a holistic representation of the semantic features and dependencies of the sentence. This research breaks new ground in two important respects. First, the model described in this paper, the Incremental Nonmonotonic Self-Organization of Meaning Network (INSOMNET), is the first subsymbolic system to be applied to deep semantic representations derived from a hand-annotated treebank of real-world sentences. Second, whereas almost all previous work has focused on the representation and learning of syntactic tree structures (such as those in the Penn Treebank), the semantic representations taken up in this study are actually dependency graphs. The challenge of developing a subsymbolic scheme for handling graph structures led to self-organizing the case-role frames that serve as the graph nodes. This semantic self-organization in turn results in a number of interesting cognitive behaviors that will be analyzed in this paper.

The INSOMNET parser combines a standard Simple Recurrent Network (SRN; Elman, 1990) with a map of input

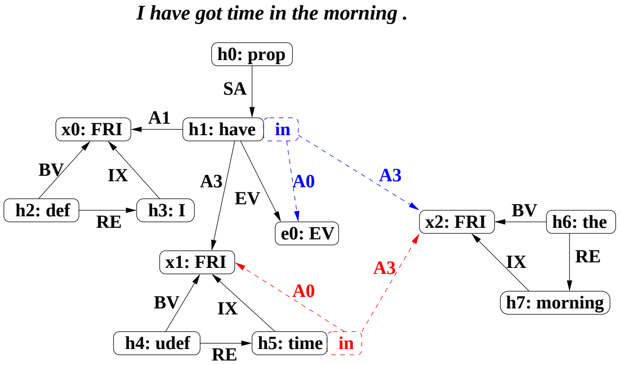

Figure 1: **MRS Dependency Graph.** This graph represents the sentence **I have got time in the morning**. The top node in the graph, labeled **h0**, has value **prop**, which tells us that this is a declarative sentence. The **have** node is the main predication of the sentence, and so serves as the *state-of-affairs* (the **SA** arc) for the **prop**. The subject (*arg1* in MRS; here **A1**) of **have** is a *full referential index* (**FRI**, with features such as gender, number, and person not shown in the figure). It is also the *instance* (**IX**) of the **I** node, and the *bound variable* (**BV**) of the determiner, **def** node, that governs the **I** node (indicated by the *restriction* arc, **RE**). Similarly, the direct object (here **A3**), of **have** is also a **FRI**, the instance of the **time** node. Third, **have** has an *event* arc **EV** that refers to an **EV** node (with features, such as aspect, mood, and tense not shown in the figure). There is one final set of nodes: **morning**, governed by the determiner **the**, also has an instance that is an object of the preposition **in**. This sentence is ambiguous: in one interpretation the preposition attaches to the verb **have** (it is **"in the morning" when I have time**), and another it attaches to the preceding noun **time** (it is **"time in the morning" that I have**). The two senses are illustrated in the figure by literally attaching a node with **in** to the **have** node and to the **time** node. Upon disambiguation, one or the other of these interpretations would be selected, but both remain coactivated until then. The distinction is made in MRS by node-sharing (attachment) and by the **in** node's **A0** arc, which points to the **EV** node in the verb-attachment case, or to the **time** node's instance **FRI** in the noun-attachment case.

words (SARDNET; Mayberry and Miikkulainen, 1999) and a novel self-organized output representation for semantic dependency graphs. The parser was trained on the Minimal Recursion Semantics (MRS; Copestake et al., 2001) representations of sentences from the LINGO Redwoods Head-driven Phrase Structure Grammar (HPSG) Treebank (Oepen et al., 2002) under development at the Center for the Study of Language and Information (CSLI) at Stanford University. This treebank incorporates deep semantic descriptions of sentences taken from the recently completed VerbMobil project (Wahlster, 2000). We report the performance of the network on the MRS dependency graphs, and illustrate its cognitive plausiblity on prepositional phrase attachment, expectations and defaults, and robustness to dysfluencies and grammatical errors in the input. The results demonstrate that subsymbolic systems can achieve incremental, nonmonotonic semantic parsing of sentences of realistic complexity.

Sentence Representation

In order to process a sentence from the Redwoods Treebank into its proper semantic representation, we need to be able to represent semantic dependency graphs. These are acyclic graphs that represent the Minimal Recursion Semantics (MRS; Copestake et al., 2001) interpretation of the sentence. MRS is a flat representation scheme where nodes represent case-role frames and arcs represent dependencies between them. Unfortunately space does not permit reviewing MRS in detail. Instead, we illustrate how MRS is used in INSOMNET by example. The MRS dependency graph for the sentence **I have got time in the morning** is shown in figure 1. This representation consists of 12 frames connected with arcs whose labels indicate the type of semantic dependency.

The MRS graph in figure 1 can be rendered as a set of frames to serve as targets for INSOMNET (figure 2). Each frame has the form *[Handle | Semantic-Relation | Subcategorization-Type | Argument-List]*. For example, the graph node labeled **h1** in the middle of figure 1 is given as the frame **[h1 | have | A1/A3/EV | x0 x1 e0]** in figure 2. The first element, **h1**, is the *Handle* (node label) of the frame: other frames can include this handle in their slots, representing a dependency to this frame. For instance, **h1** fills the *state-of-affairs* (**SA**) slot in the topmost node, **h0 prop**, as indicated by the labeled arc in figure 1 (also shown in detail in 2). The second element, **have**, gives the *Semantic-Relation* (the node value) that this frame represents. The third, **A1/A3/EV**, represents the *Subcategorization-Type* and is shorthand for the arguments that the semantic-relation takes. In this case, it indicates that **have** is a transitive verb with three arguments: subject **A1**, object **A3**, and event **EV**. The arc labels themselves are abbreviations for MRS argument names (e.g., **A1** is *arg1*, **EV** is *event*, **BV** is *bound variable*). The rest of the frame **x0 x1 e0** lists the handles (fillers) for these *arguments*. These handles refer to the other nodes in the MRS graph.

It is important to point out two main properties of handles. First, a given handle does not uniquely identify a case-role frame (node) in the MRS graph. Rather, it can be used to refer to several frames. This convention allows representing linguistic relations that are optional (in both where and whether they may occur), or that may occur more than once and therefore may require more than one frame to represent, such as adjuncts (as in the example above), modifiers (such as adjectives and relative clauses), or verb-particle constructions (e.g., "*work* something *out*"). Internally, we do use a unique designator called a subhandle (which is not part of the MRS formalism) to refer to each frame uniquely.

The second property illustrates an important difference between symbolic and subsymbolic representations. In the original MRS specification, the handles are arbitrary designators (e.g., the label **h1** has no meaning in itself). However, in the approach taken in this study, the handles are represented as patterns of activation. These patterns are learned during training so that the handles actually come to encode semantic structure. In our example, for instance, the handle **h1** actually refers to the two frames, **[have | A1/A3/EV | x0 x1 e0]** and **[in | A0/A3 | e0 x2]**. The pattern corresponding to the handle **h1** is obtained as the average of the subhandles of these two frames. The subhandle representations are in turn formed by Recursive Auto-Associative Memory (RAAM; Pollack, 1990) of the dependency graph. Starting from the semantic features at the leaves, this process allows the subhandles to be generated recursively for each node in the graph. This encoding process results in subsymbolic handles that are similar for similar structures which allows the system to generalize well to new sentences.

Network Architecture

The INSOMNET sentence parsing architecture (figure 2) consists of four components:

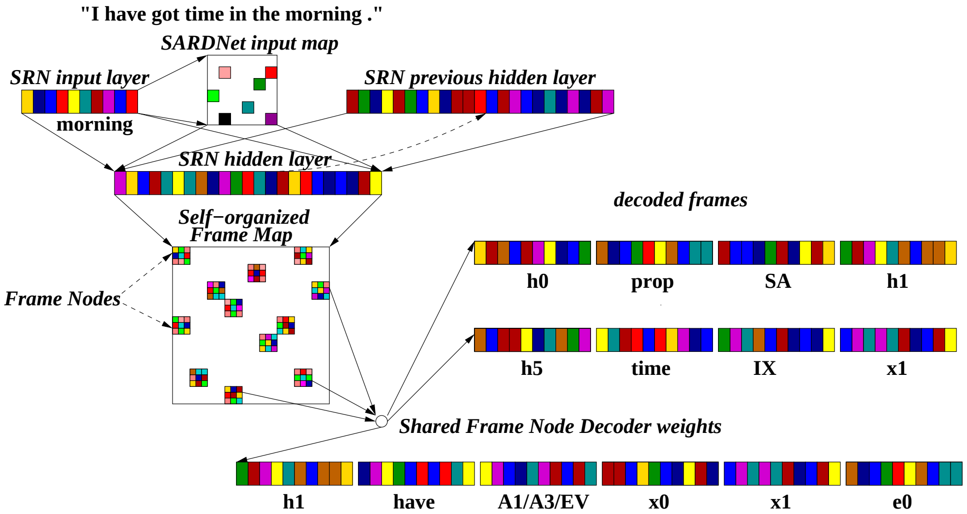

Figure 2: **The INSOMNET Architecture.** This snapshot shows the network at the end of reading the sentence **I have got time in the morning**, together with three of the decoded MRS frames corresponding to the words **have**, **time** and the topmost **prop**. The shaded units represent unit activations between 0.0 and 1.0. The SRN component reads in the sentence one word at a time. The representation for the current input word **morning** is shown at the top left. A unit corresponding to the current input word is activated on the SARDNET *input map* (at top center) at a value of 1.0 and the rest of the map is decayed by a factor of 0.9. The three output frames shown in the figure are actually decodings of the patterns (the multi-shaded squares) in the *Frame Map* (bottom center). The other patterns in the Frame Map correspond to the other nodes in the full MRS dependency graph shown in figure 1; their decodings are not shown to save space. Processing in the network proceeds as follows: as each word is read into the input buffer, it is both mapped onto the input map and propagated to the hidden layer. A copy of the hidden layer is then saved (as the *previous hidden layer*) to be used during the next time step. The hidden layer is propagated to the Frame Map, which is a 16×16 map of *Frame Nodes*, each consisting of 100 units (shown here as 3×3 patterns). The units in each Frame Node are connected through a set of shared weights that comprise the *Frame Node Decoder* to an output layer representing a case-role frame. In this way, the Frame Map can be seen as a second hidden layer. Thus, for example, the Frame Node in the top right of the map decodes into the case-role frame **[h1** | **have** | **A1/A3/EV** | **x0 x1 e0]**. The Frame Map is self-organized with the subhandles representing these case-role frame representations. Note that the argument slots and their fillers are bound together by virtue of the shared handle representation (such as **h1** between **have** and the **SA** slot of **prop**).

1. A SRN trained with BPTT to read in the input sequence.
2. A SARDNET map that retains an exponentially decaying activation of the input sequence.
3. A Self-Organized Frame Map that encodes the MRS dependency graph.
4. A Frame Node Decoder that generates the output frame representations.

The Simple Recurrent Network (SRN; Elman, 1990) is the standard neural network architecture for sequence processing, and it forms the basis for the INSOMNET architecture as well. The SRN reads a sequence of distributed word representations as input and forms the MRS dependency graph of the sentence at the output. At each time step, a copy of the hidden layer is saved and used as input during the next step, together with the next word. In this way, each new word is interpreted in the context of the entire sequence read so far, and the final parse result is gradually formed at the output. A particularly effective variant of the SRN uses backpropagation-through-time (BPTT; Williams and Zipser, 1989; Lawrence et al., 2000) to improve the network's ability to process longer sequences. With BPTT, the SRN is effectively trained as if it were a multi-layer feedforward network, with the constraint that the weights between each layer are shared.

The SARDNET is included to solve the long-term memory problem of the SRN. SARDNET is a self-organized map of word representations (James and Miikkulainen, 1995). As each word from the input sequence is read in, its corresponding unit in the map is activated at a value of 1.0, and the rest of the assembly decayed by a factor of 0.9. If an input word occurs more than once, the next closest available unit is activated. Together with the current input and previous hidden layer, the SARDNET is used as input to the hidden layer. The SARDNET identifies each input token exactly, information that would otherwise be lost in a long sequence of SRN iterations (Mayberry and Miikkulainen, 1999).

The self-organized Frame Map is the main innovation of INSOMNET. Each node in the map itself consists of a number of units. As a result of processing the input sequence, a number of these nodes will be activated, that is, a particular pattern of activation appears over the units of these nodes. Through the weights in the Frame Node Decoder, these patterns are decoded into the corresponding MRS case-role frames. The same weights are used for each node in the map. This weight-sharing enforces generalization among common elements across the many frames in any given MRS dependency graph.

The Frame Map is self-organized based on the subhandle representations. This process serves to identify which nodes in the Frame Map correspond to which case-role frames in the MRS structure. Because the subhandles are distributed representations of case-role frames, similar frames will cluster together on the map. Determiners will tend to occupy one section of the map, the various types of verbs another, nouns yet another, and so on. However, although each node becomes tuned to particular kinds of frames, no particular Frame Node is dedicated to any given frame. Rather, through different activation patterns over their units, the nodes are flexible enough to represent different frames, depending on what is needed to represent the input sequence. For example in figure 2, the

Frame Map node toward the upper right corner decodes to the **[h1** | **have** | **A1/A3/EV** | **x0 x1 e0]** case-role frame for this particular sentence. In another sentence, it could represent a different verb with a slightly different subcategorization type. This feature of the architecture makes the Frame Map able to represent semantic dependency graphs dynamically, enhancing generalization.

During training of the SRN, the Frame Node serves as a second hidden layer and the case-role frames as the output layer. The appropriate frames are presented as targets for the Frame Node layer, and the resulting error signals are backpropagated through the Frame Node Decoder weights to the Frame Node layer and on up to the first hidden layer.

At the same time, a RAAM network is trained to form the subhandle representations, and the current representations are used to organize the Frame Map. The input word representations are developed as part of the SRN training using the FGREP method (Forming Global Representations with Extended Backpropagation; Miikkulainen, 1993) and the SARDNet map is self-organized with the current representations. Eventually all these representations converge, and the networks learns to generate the correct MRS dependency graph and the corresponding case-role frames as its output.

Input Data, Training, and Experiments

The subsymbolic word representations developed by FGREP capture how the words are used in the sentences, and therefore serve as semantic representations in themselves. For this reason, the FGREP representations for the input words were used also as the fillers for semantic-relations in the MRS frames. For instance in our running example, the original semantic relations **have_rel**, **_time_mass_rel**, **_def_morning_rel**, and **_in_temp_rel** were replaced by the input words **have**, **time**, **morning**, and **in**, respectively. These changes reduced the lexicon from over 1100 tokens to just over 600. All other tokens, such as the semantic relations that do not correspond to an input word (e.g., **prop** and **def**), as well as the 40 subcategorization types (e.g., **A0/A3/EV**) and the basic semantic features that occurred in the corpus, were given random representations. All the representations (both FGREP and random) were 40-dimensional vectors between 0 and 1.

All morphemes were represented as separate tokens in the input sequence. For example, in the sentence **it look -s like i am go -ing to be pretty busy**, the morphemes **-s** and **-ing** are processed in separate steps. Such preprocessing is not strictly necessary, but it allows focusing the study on semantic processing without confounding it with morphology.

A total of 4000 sentences from the Redwoods corpus were used: 3200 for training, and the remaining 800 for testing. The shortest sentences had five frames in their MRS representation, the longest had 25. Four separate random splits of the data were used to test the INSOMNet's performance, as will be described in the next section.

Results

Figure 3 shows the average performance on the test set over the four splits measured as the proportion of fillers generated correctly. Separate plots are shown for the different MRS components, i.e., *Handles*, *Semantic-Relations*, *Subcategorization-Type*, *Features*, and *Arguments*, as well as *Null* fillers for those arguments that are not realized in the case-role frame. The main result is that the network is able to generate detailed MRS representations in its output. It performs very well on all components except semantic relations, which is not surprising since the data was more sparse with respect to semantic relations than the other components. Overall, 93% of the target MRS tokens were correctly generated, suggesting that the network had indeed learned to parse sentences into MRS dependency graphs.

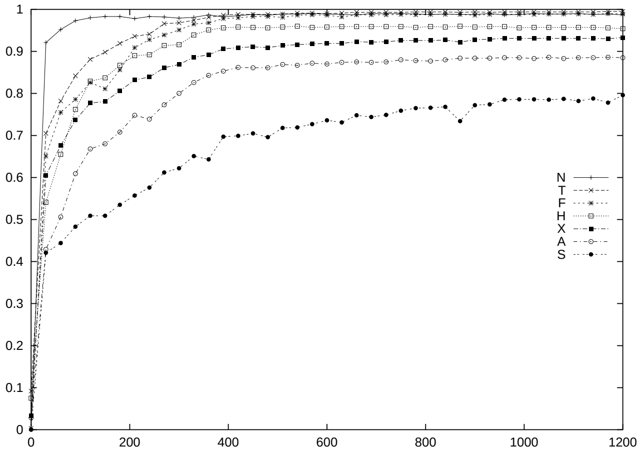

Figure 3: **Sentence Processing Performance.** The average proportion of frame constituents in the test set that were correctly produced by INSOMNet over four splits of the data during the course of training are shown here, broken down by the constituent type. The easiest for the network to learn were the arguments that had no fillers ("N"), subcategorization types ("T"), and features ("F"), all clustered near the top of the graph. The network also had little trouble generalizing the handles ("H"). More difficult were the filled arguments ("A"), and the most troublesome were the semantic ("S") representations, presumably due to their sparsity in the data. The "X" curve (black squares) gives the average of all these components. After 1200 epochs, the average performance was just over 93%. The performance on the training set was 95%, indicating that the network indeed generalizes very well.

The most interesting behavior of INSOMNet, however, takes place on top of generating the correct output in the end. It is these behaviors that make INSOMNet a potentially useful cognitive model.

First, the parsing process is incremental and nonmonotonic. As words are read in, the patterns in the Frame Map fluctuate according to the network's current interpretation as well as its expectation of how the sentence will continue. In particular, the network can revise its interpretation as it reads more of a sentence in, sometimes to the point of deactivating some frames and activating others.

Second, INSOMNet represents ambiguities explicitly, which is apparently also how humans do it in the absence of contextual clues. Several psycholinguistic studies suggest that multiple interpretations can be coactivated in parallel in the face of various types of ambiguity (Onifer and Swinney, 1981; MacDonald et al., 1992; MacDonald, 1993). Indeed, there is evidence that prepositional phrases may modify several words at the same time (Schütze, 1997). Our recurring example, **I have got time in the morning** can be used to illustrate this behavior in INSOMNet as well. Regardless of whether this sentence is new to INSOMNet (as it was

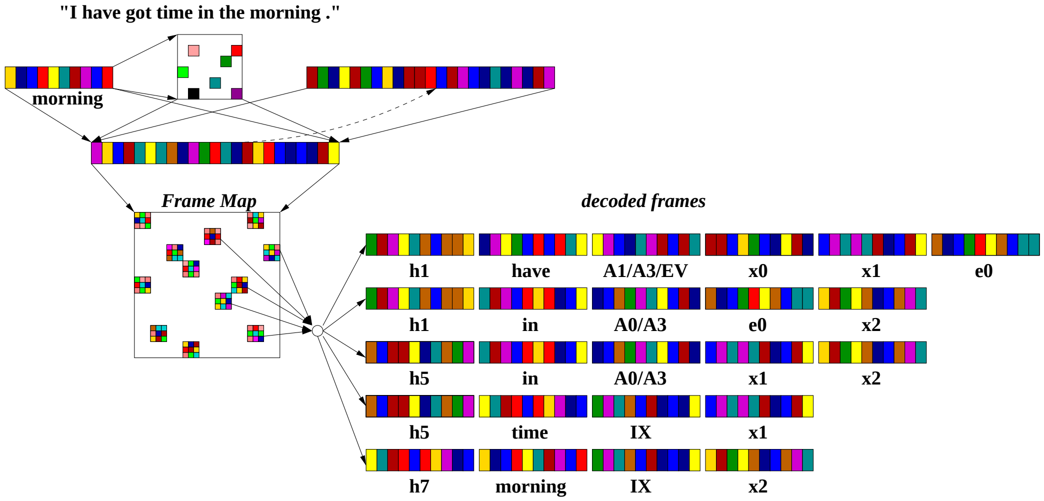

Figure 4: **Representing Ambiguity.** The sentence **I have got time in the morning** is an example of an ambiguous prepositional phrase attachment as was shown in figure 1. Both interpretations (i.e., the preposition **in** attached either to the verb **have** or to the noun **time**) are actually present in the Redwoods corpus, although in separate sentences. The network learns to exhibit both possibilities. The more likely attachment (i.e., to the verb), yields a preposition frame **in** with the same handle **h1** as the verb frame **have** to which it attaches. The other possibility is also activated: in this case the preposition frame shares a handle **h5** with the noun frame **time**. Allowing such multiple representations to be explicitly activated is one of the main advantages of the Frame Map component of INSOMNET.

in two of the splits), or INSOMNET was trained to interpret it in only one way (i.e., as a noun-attachment or a verb-attachment, as in the other two splits), it processes the sentence the same way: both possible attachments are activated in the map (figure 4). Because some sentences in the Redwoods corpus have noun-attached prepositional phrases while others have verb attachments, the network properly generalizes to represent both possibilities. This way, INSOMNET explicitly activates multiple interpretations for an ambiguous input. This behavior is cognitively valid, but has been difficult to capture in artificial parsing systems in general.

A third significant cognitive feature of INSOMNET is its robustness. A new filler, "um", not in the original lexicon and assigned a random representation, was added to all sentences in both the training and test sets at random locations. One of the networks trained on the Redwoods corpus discussed above was then tested on both these new, dysfluent sets, and performed virtually the same despite this modified input: all of the MRS case-role frames were properly generated at the output, although their activation levels were somewhat degraded in some cases. Additionally, besides the grammatical errors already present in a very few sentences in the Redwoods corpus (e.g., the sentence "here is some clues"), we've run some preliminary studies wherein we've replaced an input word with an ungrammatical variant differing in an agreement feature such as number or person, as well as deleted random articles like "a" and "the". Early results also show that the network is scarcely affected by these errors because they occur so infrequently compared to its training history. These results suggest that INSOMNET can tolerate noisy, dysfluent, and ungrammatical input much like people do.

Fourth, the network demonstrates expectations and defaults which have become a hallmark of subsymbolic systems. Because the network is trained to output the full representation of the MRS semantic dependency graph, it learns to anticipate certain frames before they have been licensed by the input. Similarly, the network exhibits defaults and even semantic illusions: when it misses a component in a frame, it will substitute a more frequent analogue. Both expectations and semantic illusions are common in human natural language understanding and arise automatically in the INSOMNET model.

Discussion and Future Work

The ultimate goal of this research is to develop a subsymbolic parser that can handle realistic language without sacrificing those characteristics of neural networks that make them powerful cognitive models. The described method of representating MRS dependency graphs permits the network to gradually refine its output to accommodate changes as new information comes in. In this paper, we have shown that this behavior can be preserved while scaling up to the realistic linguistic structures present in the LINGO Redwoods Treebank.

Our future work focuses on three further important developments of INSOMNET. First, we will augment the model with a gating mechanism that modulates the activations of the Frame Node patterns. Preliminary experiments show that this mechanism dramatically enhances the nonmonotonic behavior of INSOMNET. In particular, gating suppresses the activations of Frame Nodes that should not be a part of the MRS dependency graph while at the same time providing a soft threshold for relevant nodes. These experiments also indicate that gating also accentuates coactivation of multiple interpretations, as well as expectations and defaults, which will allow a more quantitative assessment of these behaviors.

Second, we will replace the tokens in the input with either orthographic or phonological representations. The strong tendency of INSOMNET to create expectations and its general robustness should then allow it to process unknown words systematically. At the same time, the network should also learn to identify morphological components in its input representations and map them onto their proper semantic targets, removing the need for preprocessing the input data.

Third, we plan to test INSOMNET as a robust system for parsing spoken language. They system will be trained with the actual transcripts in the VerbMobil corpus, which include dysfluencies of everyday spoken language, such as false starts, repairs, hesitations, and fillers. We expect the system to learn their structure, and to learn to compensate for them in the sentence interpretation. If so, INSOMNET could serve as a significant step towards scaling up semantics parsing to the real world.

Conclusion

In this paper, we presented a subsymbolic parser, INSOMNET, that is able to parse a real-world corpus of sentences into semantic representations. A crucial innovation was to use an MRS dependency graph as the sentence representation, encoded in a self-organized Frame Map. As is typical of holistic parsers, the parse result is developed nonmonotonically in the course of incrementally reading in the input words, thereby demonstrating several cognitive behaviors such as coactivation, expectations and defaults, and robustness. These properties make INSOMNET a promising foundation for understanding human sentence processing in the future.

References

Allen, J. and Seidenberg, M. S. (1999). The emergence of grammaticality in connectionist networks. In MacWhinney, B. J., editor, *Emergence of Language*, pages 115–151. Erlbaum, Hillsdale, NJ.

Berg, G. (1992). A connectionist parser with recursive sentence structure and lexical disambiguation. In Swartout, W., editor, *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 32–37. Cambridge, MA: MIT Press.

Christiansen, M. H. and Chater, N. (1999). Toward a connectionist model of recursion in human linguistic performance. *Cognitive Science*, 23(2):157–205.

Copestake, A., Lascarides, A., and Flickinger, D. (2001). An algebra for semantic construction in constraint-based grammars. In *Proceedings of ACL-2001*. Toulouse, France.

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

Hammerton, J. (2001). Clause identification with long short-term memory. In Daelemans, W. and Zajac, R., editors, *Proceedings of CoNLL-2001*, pages 61–63.

Ho, E. K. S. and Chan, L. W. (2001). Analyzing holistic parsers: Implications for robust parsing and systematicity. *Neural Computation*, 13(5):1137–1170.

James, D. L. and Miikkulainen, R. (1995). SARDNET: A self-organizing feature map for sequences. In Tesauro, G., Touretzky, D. S., and Leen, T. K., editors, *Advances in Neural Information Processing Systems 7*, pages 577–584. Cambridge, MA: MIT Press.

Lane, P. C. R. and Henderson, J. B. (2001). Incremental syntactic parsing of natural language corpora with simple synchrony networks. *IEEE Transactions on Knowledge and Data Engineering*, 13(2):219–231.

Lawrence, S., Giles, C. L., and Fong, S. (2000). Natural language grammatical inference with recurrent neural networks. *IEEE Transactions on Knowledge and Data Engineering*, 12(1):126–140.

MacDonald, M. C. (1993). The interaction of lexical and syntactic ambiguity. *Journal of Memory and Language*, 32:692–715.

MacDonald, M. C., Just, M. A., and Carpenter, P. A. (1992). Working memory constraints on the processing of syntactic ambiguity. *Cognitive Psychology*, 24:56–98.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English: The Penn treebank. *Computational Linguistics*, 19:313–330.

Mayberry, III, M. R. and Miikkulainen, R. (1999). Using a sequential SOM to parse long-term dependencies. In *Proceedings of the 21st Annual Conference of the Cognitive Science Society*, pages 367–372. Hillsdale, NJ: Erlbaum.

McClelland, J. L. and Kawamoto, A. H. (1986). Mechanisms of sentence processing: Assigning roles to constituents. In McClelland, J. L. and Rumelhart, D. E., editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 2: Psychological and Biological Models*, pages 272–325. MIT Press, Cambridge, MA.

Miikkulainen, R. (1993). *Subsymbolic Natural Language Processing: An Integrated Model of Scripts, Lexicon, and Memory*. MIT Press, Cambridge, MA.

Miikkulainen, R. (1997). Dyslexic and category-specific impairments in a self-organizing feature map model of the lexicon. *Brain and Language*, 59:334–366.

Oepen, S., Flickinger, D., Toutanova, K., and Manning, C. (2002). LinGO redwoods: A rich and dynamic treebank for HPSG. In *Beyond PARSEVAL. Workshop of the Third LREC Conference*.

Onifer, W. and Swinney, D. A. (1981). Accessing lexical ambiguities during sentence comprehension: Effects of frequency of meaning and contextual bias. *Memory and Cognition*, 9:225–226.

Plaut, D. C. and Shallice, T. (1993). Perseverative and semantic influences on visual object naming errors in optic aphasia: A connectionist account. *Journal of Cognitive Neuroscience*, 5(1):89–117.

Pollack, J. B. (1990). Recursive distributed representations. *Artificial Intelligence*, 46:77–105.

Rohde, D. T. (2002). *A Connectionist Model of Sentence Comprehension and Production*. PhD thesis, Computer Science Department, Carnegie Mellon University, Pittsburgh, PA. Technical Report CMU-CS-02-105.

Schütze, H., editor (1997). *Ambiguity Resolution in Language Learning: Computational and Cognitive Models*. CSLI, Stanford University.

Sharkey, N. E. and Sharkey, A. J. C. (1992). A modular design for connectionist parsing. In Drossaers, M. F. J. and Nijholt, A., editors, *Twente Workshop on Language Technology 3: Connectionism and Natural Language Processing*, pages 87–96, Enschede, the Netherlands. Department of Computer Science, University of Twente.

St. John, M. F. and McClelland, J. L. (1990). Learning and applying contextual constraints in sentence comprehension. *Artificial Intelligence*, 46:217–258.

Wahlster, W., editor (2000). *Verbmobil. Foundations of Speech-to-Speech Translation*. Springer-Verlag, Berlin; New York.

Williams, R. J. and Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1:270–280.