Semantic Interpretation as Classificatory Abduction

Julie Hartigan
The Ohio State University
The Laboratory for Artificial Intelligence Research
Department of Computer and Information Science
Columbus, Ohio 43210 USA
(614) 292-1015
Email: harti@cis.ohio-state.edu

Abstract

This paper proposes a new computational framework, called classificatory abduction, to perform some aspects of natural language understanding. Inherent in its computational design is the ability to cope with the intractability of semantic interpretation.

This work has been supported by an Ohio State University Interdisciplinary Research Grant.

Semantic Interpretation as Classificatory Abduction

Abstract

This paper proposes a new computational framework, called classificatory abduction, to perform some aspects of natural language understanding. Inherent in its computational design is the ability to cope with the intractability of semantic interpretation.

1 Introduction

Classification and abduction are domain independent computational strategies for which generic control strategies and knowledge representations can be specified. Classification's knowledge organization enables large-scale pruning of a search space. Abduction produces a conclusion given data to explain. The task of natural language understanding can be viewed as a combination of these two techniques into a new problem-solving strategy called classificatory abduction.

In this paper I will briefly discuss the use of classification and abduction individually with respect to natural language understanding, and then present classificatory abduction. I will conclude with an example of semantic interpretation as classificatory abduction.

2 Approaches to NLU

2.1 Classification

Classification is the process of finding the categories or hypotheses within a classification hierarchy that apply to the situation being analyzed [Bylander and Mittal, 1986]. The goal is to output the most specific hypothesis or hypotheses which are plausible given the input data. The hierarchical structure enables class-subclass knowledge to be easily expressed. Each node in the hierarchy represents a hypothesis, with the more general hypotheses closer to the root node, and the most specific hypotheses as leaves. A generic control strategy for classification is Establish-Refine [ibid.], where if a hypothesis (node in the hierarchy) is plausible, it establishes. Once established, it tries to refine itself by establishing any of its children. A node which finds its hypothesis to be implausible rules itself out which causes the entire subtree beneath it to also be ruled out.

Systemic grammar is one example of a classificatory description of natural language. Winograd (1983) contains an excellent synopsis of systemic grammars. An example of a computational systemic grammar is Nigel, which was at the heart of the Penman text generation system [Mann and Matthiessen, 1983]. To find the correct path through its hierarchy, each node had an expert that asked questions to determine which of its children should establish. A module of code was also written that could answer questions based on the goals of the speaker, discourse issues, etc. This system's success at generation shed great hope on its potential at understanding. However, to accomplish this task, the classification hierarchy (systemic grammar) was translated into a functional unification grammar [Kasper, 1988]. We would like to devise a framework in which systemic hierarchies can be used directly in the process of natural language understanding.

2.2 Abduction

The information processing task of abduction is inference to the best explanation. Given a set of data, abduction infers a hypothesis, or set of hypotheses called a composite, that best accounts for the data [Josephson *et al.*, 1987]. What is meant by *best* is that the composite will be parsimonious, maximally plausible, explanatorily complete, and consistent. Other factors that are taken into account are: how good the composite hypothesis is by itself, how much better it is than the alternatives, how much confidence there is in the accuracy of the data, and how thorough the search was for alternative explanations.

Natural language understanding is an abductive task [Charniak and McDermott, 1985]. One implementation that utilized abduction as the problem-solving strategy for natural language understanding was created by Hobbs *et al.* (1990). They use predicate logic as their form of knowledge representation and abduce over the propositions and predicates. Their abductive inference is computationally intractable [Bylander *et al.*, 1990].

Abduction has been implemented in many ways. In general, the hypothesis that explains a datum that no other hypothesis can explain must be included in the composite; it is *essential* to the explanation. Next, the hypothesis that explains a datum better than other hypothesis and with greater plausibility than a certain threshold value are included in the composite. Then, the composite must be checked for consistency (one hypothesis in the composite cannot rule out another) and made parsimonious. These steps are continued until all the data can be explained or until there is "sufficient" explanatory coverage. Abduction, in general, is intractable. Therefore, it is necessary to reduce the number of hypotheses under consideration at any given time to make the algorithm efficient.

2.3 Classificatory Abduction

The approach that I am promoting is the use of *classificatory abduction*. This combines the knowledge organization and plausible hypothesis generation ability of classification with the problem-solving strategy of abduction to attain a system that can efficiently arrive at a conclusion. This computational framework has been used successfully in the domain of medical diagnosis [Josephson *et al.*, 1985], but has been relatively unexplored in the domain of natural language understanding.

Classification hierarchies — in the form of systemic grammars — are used to encode hierarchies of linguistic knowledge. The prototype system that performs classificatory abduction employs a stratified hierarchical approach, assigning a stratum to represent socio-pragmatics (the intended meaning) and another stratum to represent the logicogrammatical information (semantics, grammar, and some discourse information). Both strata are arranged as classification hierarchies and a mapping is defined between them [Winograd, 1983]. The socio-pragmatic stratum has features that represent both pragmatic and sociolinguistic information. The basic computational advantage of including the socio-pragmatic stratum is a person's expectations of what may be said can provide additional guidance through the hierarchy. Therefore, not all the processing is bottom-up, but some processing is performed top-down. In essence, we are bringing more knowledge to bear on the problem in order to decrease the abductive processing and hence increase the efficiency. The system also benefits in that it has the hypotheses arranged hierarchically, therefore the number of hypotheses being considered at any one time is drastically reduced. In fact, classificatory abduction is based on an abduction technique that was proven to be tractable [Allemang *et al.*, 1987].

Since research has been completed that utilizes this same systemic grammar for the purpose of natural language generation [Patten *et al.*, 1992], the grammar is bidirectional.

The input to the system is the systemic grammar and a structural analysis of the sentence. The structural information is assumed to be fact, and hence those features in the classification hierarchy that match the structural data are deemed *essential* to the explanation, or the understanding, of the sentence. The status of *essential* is propagated up through the parent nodes in the systemic classification hierarchy, since an *essential* child implies an *essential* parent. Also, any constraint imposed by an *essential* feature on a daughter constituent (e.g. by a clause feature on a noun phrase feature) is *essential*. All *essential* features are included in the composite. Once the essentials phase is complete, the socio-pragmatic expectations are instantiated, causing more logicogrammatical features to be added to the composite. The next step is to make decisions between competing siblings. If one child is a default feature, it is put in the composite first. Default features are those that are assumed to be true until proven otherwise. Whenever a node is put in the composite, its realization rules are posted as constraints. Realization rules define the structure of the sentence in terms of surface and deep roles and are represented in Figure 1. by the shaded boxes. When an inconsistency occurs among these constraints, backtracking takes place. The current prototype system uses chronological backtracking rather than dependency-directed backtracking. Additional hypotheses are placed in the composite until all the input is accounted for and a complete path is made through the hierarchy.

3 Example

A prototype system has been developed that can perform semantic analysis on a variety of sentences. Here is an example of its performance on one sentence. The relevant socio-pragmatic information is described in Patten *et al*. (1992). Figure 1. is a minute fragment of the actual systemic grammar.

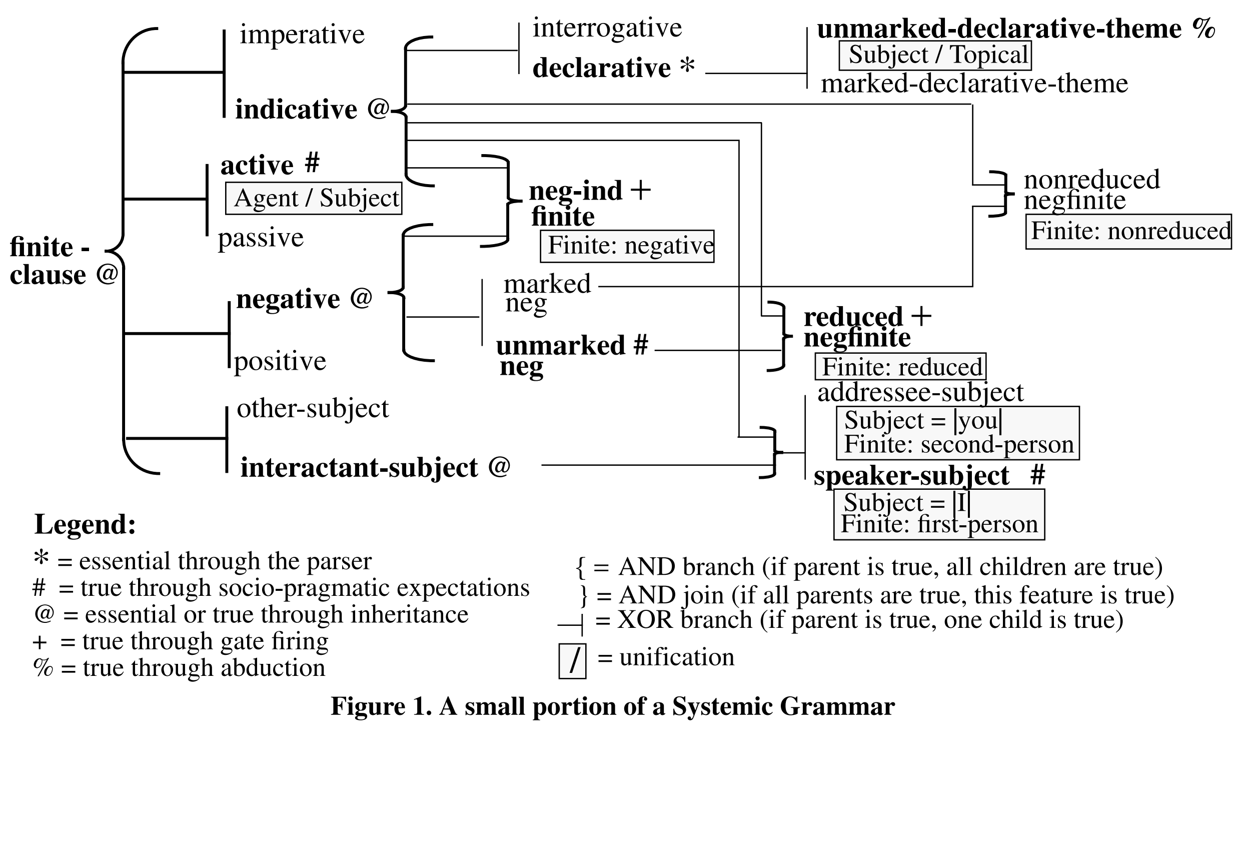

Figure 1. A small portion of a Systemic Grammar

Assume a scenario where a person has called a travel agent and asked for a seat on a popular flight, hence the person knows the chances of getting on that flight are slim. The travel agent says, “I am not finding anything then”. Assume the parser provides the information that the sentence is *declarative*, therefore the feature *declarative* is essential. *Indicative* and *finite-clause* become essential since they are predecessors to *declarative.* Next, the expectations from the pragmatic stratum, *unsuccessful-so-far* and *agent-oriented*, are instantiated. These features cause *unmarked-neg*, *active*, and *speaker-subject* to be true and includes them in the composite. Their predecessors (*interactant-subject* and *negative*) are also added to the composite. The realization rules are posted to check for inconsistencies, but none occur. The AND joins are evaluated, which adds *reduced-negfinite* and *neg-ind-finite* to the composite. At this point, nothing more can be determined with the available knowledge. A decision must be made as to whether the sentence is *marked-* or *unmarked-declarative-theme*. Since the *unmarked-declarative-theme* is the default feature, it is included in the composite. The composite is checked for inconsistencies, and none exist.

At this point, a complete path has been abduced using the structural analysis and the expectations from the pragmatic stratum. If the pragmatic stratum had not been used to guide the processing, then the determination of the subject, voice, and polarity of the sentence would have required additional processing time, hence decreasing the overall system efficiency. Also, by using the classification hierarchy, putting a feature into the composite ruled out that feature’s siblings and all of their children.

4 Conclusion

The advantages of this framework are:

Efficiency: By using systemic grammar as the knowledge representation, we have a hierarchy of hypotheses. Ruling out one hypothesis rules out all of that hypothesis’ descendants. Further efficiency is attained by mapping contextual information directly to features in the systemic grammar, hence reducing the number of hypotheses needing to be considered.

Sensitivity to Social/Contextual Issues: The socio-pragmatic knowledge available in the context can, in principle, be exploited during the semantic interpretation.

Bidirectional: This system uses the same grammar as Patten *et al.* (1992) used for generation. Hence, this grammar is independent of its use, and can be used for both understanding and generation.

Classification and top-down guidance are promising techniques for dealing with the intractability of the abductive inference inherent in semantic interpretation.

References

[Allemang *et al*., 1987] D. Allemang, M. Tanner, T. Bylander, and J Josephson. On the Computational Complexity of Hypothesis Assembly. In *Proc. Tenth Int. Joint Conf. on Artificial Intelligence*, pp. 1112-1117, Milan.

[Bylander and Mittal, 1986] T. Bylander and S. Mittal. CSRL: A Language for Classificatory Problem Solving and Uncertainty Handling. *AI Magazine,* 7(3):66-77.

[Bylander *et al*., 1990] T. Bylander, D. Allemang, M. Tanner and J. Josephson. The Computational Complexity of Abduction. Tech. Report. OSU.

[Charniak and McDermott, 1985] E. Charniak and D. McDermott. *Introduction to Artificial Intelligence*. Addison-Wesley Publishing Company, Reading MA.

[Hobbs *et al*., 1990] J. R. Hobbs, M. Stickel, D. Appelt, and P. Martin. Interpretation as Abduction. Technical Report 499, SRI International, Menlo Park, CA.

[Josephson *et al*., 1985] J. R. Josephson, M. C. Tanner, J. W. Smith, M.D., J. Svirbely, and P. Straum. Red: Integrating Generic Tasks to Identify Red-Cell Antibodies. In *Proc. of The Expert Systems in Government Symposium*, McLean, VA.

[Josephson *et al*., 1987] J. R. Josephson, B. Chandrasekaran, J. W. Smith, and M. C. Tanner. A Mechanism for Forming Composite Explanatory Hypotheses. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-17(3):445-454.

[Kasper, 1988] R. Kasper. Parsing with a Systemic Grammar. Presented at the 15th International Systemic Congress.

[Mann and Matthiessen, 1983] W. Mann and C. Matthiessen. Nigel: A Systemic Grammar for Text Generation. ISI/RR-83-105, USC. pp. 1-55.

[Patten *et al*., 1992] T. Patten, M. Geis, and B. D. Becker. Toward a Theory of Compilation for Natural Language Generation. *Computational Intelligence*, 8(1):77-101.

[Winograd, 1983] T. Winograd. *Language as a Cognitive Process*. Addison-Wesley, London.