Maintaining Consistency and Plausibility in Integrated Natural Language Understanding

Toyoaki Nishida, Xuemin Liu, Shuji Doshita, and Atsushi Yamada
Department of Information Science
Kyoto University
Sakyo-ku, Kyoto 606, Japan
phone: 81-75-751-2111 ext. 5396
e-mail: nishida@doshita.kuis.kyoto-u.ac.jp, relay.cs.net

Abstract

In this paper, we present an inference mechanism called the integrated parsing engine which provides a uniform abductive inference mechanism for natural language understanding. It can (1) make plausible assumptions, (2) reason with multiple alternatives, (3) switch the search process to the maximally plausible alternative, (4) detect contradiction and tame conclusions which depend on inconsistent assumptions, and (5) update plausibility factor of each belief based on new observations. We demonstrate that a natural language understanding system using the integrated parsing engine as a subsystem can pursue a guided search for most plausible interpretation by making use of syntax, semantics, and contextual information.

1 Introduction

Natural language understanding involves lots of hard issues such as various types of ambiguities, indeterminacies caused by ellipses or fragmental utterances, or ill-formedness. Being confronted with these difficulties, it does not seem reasonable to seek for a method of logically deducing the speaker’s intended meaning or plan from utterances. Instead, it is much more natural to characterize natural language understanding as an abductive process of exploring most plausible interpretation which can explain given utterances.

In this paper, we present an abductive inference mechanism, called the integrated parsing engine, for natural language understanding. The integrated parsing engine is able to:

- make plausible assumptions at appropriate time
- reason with multiple alternatives based on different sets of assumptions
- switch the search process to the maximally plausible alternative
- detect contradiction resulting from inconsistent assumptions and eliminate all conclusions which depends on these assumptions
- update plausibility factor of each belief based on new observations.

Thus, the integrated parsing engine is general enough to carry out linguistic and nonlinguistic inferences in a uniform manner, by drawing information from various sources: syntax, semantic, discourse, pragmatics, or real world.

In the remainder of this paper, we first describe mechanisms for maintaining consistency and plausibility. We then show how these two mechanisms interact to guide the inference process. Finally, we use an implemented example to demonstrate how the integrated parsing engine is used to interpret sentences by taking contextual factors into account.

2 Maintaining Consistency

The CME (Consistency Maintenance Engine) is a component of the integrated parsing engine
responsible for maintaining consistency among beliefs. Basic design principles of the CME is based on de Kleer’s ATMS (Assumption-based Truth Maintenance Engine) [de 86]. The CME maintains a set of alternative beliefs, each of which consists of a set of assumptions and their conclusions, as follows:

- alternative 1: \{A_{11}, \ldots, A_{1m_1}\} B_{11}, \ldots, B_{1m_1}
- alternative n: \{A_{n1}, \ldots, A_{nm_n}\} B_{n1}, \ldots, B_{nm_n}

An external problem solver is assumed to exist which makes assumption, adds conclusion, and detects contradiction.

The main task of CME is to maintain alternative beliefs by removing all alternatives whose set of assumptions has turned out contradictory. Like ATMS, the CME takes advantage of the following monotonic property:

If contradiction is derived from a set of assumptions \(A\), then contradiction is also derived from any set of assumptions \(B\) such that \(B \supseteq A\).

Thus, if contradiction is derived from a set of assumptions \(\{B, D\}\), alternative interpretations depending on sets of assumptions such as \(\{B, C, D\}\), \(\{A, B, D\}\), \(\{A, B, C, D\}\), \ldots are removed. In addition, the CME keeps records of contradictory sets of assumptions to prevent any interpretation depending on them from being considered in future.

Unlike ATMS whose control regime is breadth-first, our CME uses a tree called the environment tree, or the E-tree for short, to guide the search process. Each node of the E-tree represents an environment, a set of assumptions. Each arc of the E-tree represents that a lower node is derived from the upper node by making one more assumption. Thus in figure 1, \(E_0\) is the root node, and it represents an environment without any assumption. Nodes below \(E_0\) represent environments with one or more assumption added to its parent node’s environment. Thus, \(E_1 = E_0 \cup \{A_1\} = \{A_1\}\), \(E_{11} = E_1 \cup \{A_{11}\} = \{A_1, A_{11}\}\), and so on.

We assume that a set of assumptions made at the same parent node are mutually exclusive. Although this is a rather strong assumption, it makes sense in natural language understanding since many assumptions being made during the natural language understanding process are mutually exclusive. Even if this is not the case, any set of assumptions can be transformed into a set of mutually exclusive assumptions by adding appropriate conditions. Although this is a cumbersome solution, it does not often take place in natural language understanding and most importantly it saves the amount of computation.

Note that the CME alone cannot determine which way to go when there is more than one possibility of extending the set of beliefs. This information is provided by the PME, as described in the next section.

### 3 Maintaining Plausibility

The PME (Plausibility Maintenance Engine) maintains estimations of how plausible each environment is. This information is given as conditional probabilities and it is kept as annotations to each arc of the E-tree. Thus, in figure 2, which is a slightly more precise version of figure 1, \(p_i\) stands for \(P(E_i)\), \(p_{ij}\) for \(P(E_j|A_i)\), \(p_{ijk}\) for \(P(E_k|A_i, A_j)\), etc.

It follows from the property of conditional probability that

\[
P(E_i|\ldots E_j\ldots) = 0,
\]

if \(i \neq j\) and \(E_i\) and \(E_j\) are immediate children.
Figure 3: A Sample E-tree with Annotation of the same parent. Furthermore,

\[ P(E_i \ldots \neg E_j \ldots) = 0, \]

if \( E_j \) is a parent node of \( E_i \).

Initial value of \( p_i \)'s are to be given from the external problem solver. The PME's role is to maintain estimation of prausibility by taking into account given observations. Currently we only take \( \neg E \), the event of environment \( E \) running into contradiction, as an observation. We use a Bayes' law to modify \( P(A) \) into \( P(A|\neg E) \). Thus,

\[
P(E_i|\neg E_j) = \frac{P(\neg E_j|E_i) \cdot P(E_i)}{P(\neg E_j)} = \frac{(1 - P(E_j|P_i)) \cdot P(E_i)}{1 - P(E_j)}. \tag{1}
\]

if \( E_i \) and \( E_j \) are brothers, (1) is further simplified to:

\[
P(E_i) \frac{1 - P(E_j)}{1 - P(E_j)|}. \tag{2}
\]

For example, suppose it has turned out that environment \( E_4 \) is in contradiction and hence \( \neg E_4 \) is observed (figure 3(a)). The annotations to the E-tree are updated as in figure 3(b). Notice that the update of conditional probability can be done based on local information.

4 Natural Language Understanding System Using the Integrated Parsing Engine as a Subsystem

The integrated parsing engine consists of the CME and the PME. The architecture of a natural language understanding system with the integrated parsing engine as a subsystem is shown in figure 4.

The knowledge base contains various types of information for language comprehension, including lexicon, morphology, syntax, semantics, discourse, pragmatics, commonsenses, and so on. The whole system is controled by the problem solving engine (PSE). The PSE can access to the knowledge base and use the integrated parsing engine as an aid to seek for most plausible interpretation. Input texts are analyzed in a sentence-by-sentence manner. The discourse structure is maintained as a previous topic in the working memory.

When it scans a new sentence, the PSE first initialize the E-tree with only the root node. Then the PSE repeats the following cycle:

(step 1) choose a leaf node with the highest probability as a working environment

(step 2) repeatedly derive conclusions from
believed propositions until either (a) the goal is achieved, (b) contradiction is derived, or (c) no more conclusion is derived unless making more assumption.

In case (a), the process halts.

In case (b), the process is passed to the PME, which modifies current estimation of plausibility so that this fact is reflected, then an alternative of maximum plausibility is chosen and is suggested to the CME.

In case (c), the process also is passed to the PME, which assigns plausibility to new nodes, and working environment is chosen again.

The integrated parsing engine has been written in Lisp. It is running with a small experimental grammar for Japanese. The next section shows how it works.

5 An Example

Suppose a dialog environment in which a professor speaks to a clerk to borrow a key of some rooms (figure 5) and utters the following Japanese sentence:

(3) KAGI WO KASHITE KUDA SAI
   (a/the) key <object> lead could you ...?
   "could you lead (me) (a/the) key?"

The referential meaning of this sentence is ambiguous if there is more than one key in a given situation. Suppose three keys are there: key1 for a library room, key2 for a xerox room, and key3 for a meeting room.

Although sentence (3) is ambiguous in normal contexts, it becomes much less so if it follows sentences like:

(4) HO N WO PI I SHITA I NO DE SU GA
    "I'd like to xerox some books."

Even if no previous sentence is spoken, sentence (3) is acceptable in a situation where the speaker and the hearer mutually believe that the xerox room is accessed so often that "the key" is usually used to refer to key2, the one for the xerox room.

Note that the omission of the patient case does not matter in usual situations, since there is a strong default that the filler of this case is the speaker.

Now let us show how sentence (3) is analyzed in a context where sentence (4) was previously uttered. The task of analyzing input starts from recognizing words. Lots of ambiguities arise in this phase. For sentence (3), ‘KA’ might be a single word ‘KA’ (postposition marking interrogative) or a part of a longer word ‘KAGI’ (key). Since longer match is considered to be more plausible in general case in Japanese analysis, we assign larger number of probability to the latter possibility. Following this analysis, the PSE makes the assumptions to the integrated parsing engine:

- @word-1 (take the sequence ‘KA’ as a word):
  \(\Rightarrow\) probability 1/3.

- @word-2 (take the sequence ‘KAGI’ as a word):
  \(\Rightarrow\) probability 2/3.

Accordingly, the CME extends the initial E-tree as in figure 6. Since, the environment \(E_1\) has the highest plausibility, the CME chooses it for the next environment and control is returned to the PSE.
Now the PSE tries to derive further conclusion in the chosen environment. After having recognized that the part of speech of the word 'KAGI' is noun, the PSE tries to find out the referent of the noun and realizes that three ambiguities arise in this situation. Again, the PSE calls the CME to make assumptions. At the same time, the PSE is called for to assign estimated conditional probabilities to each assumption.

Currently, the system uses an associative network as shown in figure 7 to determine plausibility. Nodes of this network represent either a concept or an instance, and arcs mean that the two concepts or instants at its both ends have a certain relation. Those items which have dense connections to previous subjects are considered to be plausible as a referent. In our example, since the node xerox is marked as the previous subject key2 is considered most plausible, while key1 is less plausible and key3 much less. Thus, the following assumptions are made: 1

@referent-1 (consider 'KAGI' to refer to key1):
⇒ probability 1/3.
@referent-2 (consider 'KAGI' to refer to key2):
⇒ probability 1/2.
@referent-3 (consider 'KAGI' to refer to key3):
⇒ probability 1/6.

In case no previous utterance is given, the PSE will consult information given as a priori measurements.

The E-tree now becomes as in figure 8, and {@word-2, @referent-2}, which is the most plausible environment at this point, is chosen as the next environment. The analysis is continued this way until the semantic representation is obtained for the whole sentence. The interpretation obtained this case is:

[ event = asking-for
actor = <the speaker>
object = key2 ]

Figure 9 shows the dependency structure of beliefs related to this analysis.

Notice that the efficiency of the analysis is significantly improved when strong expectation exists. For example, although character 'SHI' in sentence (3) has many possible interpretations in Japanese, the system is not annoyed by those ambiguities, since this part of the sentence just goes as expected. The system may come to suspect it only when most of its expectation fails.

1Currently we use a very simple algorithm for assigning those value: when there are three alternatives, the densest connection receives the value (1/3), the second (1/2), and the third (1/6), regardless of how closely they are related to each other. We plan to develop a much more precise method in a near future.
addition, the integrated parsing engine provides a concise and high level mechanism for abductive reasoning. We have carefully chosen a set of reasonably high-level functions necessary for abductive reasoning. This serves to much simplifying natural language understanding system than otherwise.

7 Concluding Remarks

We have presented an inference engine for integrated natural language understanding, based on a characterization of natural language understanding as an abductive process. The essence of our approach is connecting consistency maintenance engine and plausibility maintenance engine closely enough to allow their dense interaction. Although we have shown rather "low level" issues, we believe the same idea is applicable to "higher level" problems such as inferring speaker's intention and plan.

References

[Cha86] Engine Charniak. A neat theory of marker passing. In Proceedings AAAI-86, pages 584–588, 1986.
[de 86] Johan de Kleer. An assumption-based tms. Artificial Intelligence, 28:127–162, 1986.
[Lyt86] Steven Lytinen. Dynamically combining syntax and semantics in natural language processing. In Proceedings AAAI-86, pages 574–578, 1986.
[WP85] D. Waltz and J. B. Pollack. Massively parallel parsing. Massively interactive model of natural language interpretation. Cognitive Science, 9:51–74, 1985.