

# Using Parsed Corpora for Structural Disambiguation in the TRAINS Domain

Mark G. Core and Lenhart K. Schubert

The University of Rochester  
Computer Science Department  
Rochester, New York 14627

Technical Report 608

February 1996

## Abstract

This paper describes a prototype disambiguation module, KANKEI, which uses two corpora of the TRAINS project. In ambiguous verb phrases of form V...NP PP or V...NP adverb(s), the two corpora have very different PP and adverb attachment patterns; in the first, the correct attachment is to the VP 88.7% of the time, while in the second, the correct attachment is to the NP 73.5% of the time. KANKEI uses various n-gram patterns of the phrase heads around these ambiguities, and assigns parse trees (with these ambiguities) a score based on a linear combination of the frequencies with which these patterns appear with NP and VP attachments in the TRAINS corpora. Unlike previous statistical disambiguation systems, this technique thus combines evidence from bigrams, trigrams, and the 4-gram around an ambiguous attachment. In the current experiments, equal weights are used for simplicity but results are still good on the TRAINS corpora (92.2% and 92.4% accuracy). Despite the large statistical differences in attachment preferences in the two corpora, training on the first corpus and testing on the second gives an accuracy of 90.9%. These results suggest that our technique captures attachment patterns that are useful across corpora.

---

This work was supported in part by National Science Foundation grant IRI-95033312. James Allen's helpful suggestions are gratefully acknowledged.

# 1 Introduction

The goal of the TRAINS project is to build a computerized planning assistant that can interact conversationally with its user. The current version of this planning assistant, TRAINS 95, is described in [AFMR95]; it accepts speech input and aims primarily at robustness. The planning problems handled involve moving several trains from given starting locations to specified destinations on a map display (showing a network of rail lines in the eastern United States). The 1995 TRAINS system uses a relatively simple grammar to help achieve robustness. However, the grammar still allows attachment of some adverb and prepositional phrase (PP) postmodifiers to either noun phrases (NPs) or verb phrases (VPs). The current parser always attaches PPs and adverbs to the NP instead of the VP.

This strategy works sufficiently well for the simple planning dialogs currently handled by the TRAINS 95 system. One reason the dialogs are simple is that the problems and domain are themselves simple. But once the complexity of the problems is increased, the parser will need a disambiguation system. A corpus (the 95 dialogs) was created from people's utterances to the TRAINS 95 system. A sample from this corpus with the system's utterances inserted is shown in figure 1.

```
1.1a) USER: OKAY I WANT TO GO FROM COLUMBUS NO I WANT TO GO TO COLUMBUS
        CLEVELAND AND RICHMOND
1.1b) SYSTEM: Nope. Sorry, but I can't seem to find a train at Columbus.
        I need to know an engine
1.2a) USER: NO I WANT TO GO TO COLUMBUS CLEVELAND AND RICHMOND
1.2b) SYSTEM: I can see you are trying to get to City Columbus, but you need
        to tell me with what. Yeah.
1.3a) USER: USE THE ENGINE AT CHARLOTTE
1.3b) SYSTEM: Huh?
1.4a) USER: USE THE ENGINE AT CHARLOTTE TO GO TO COLUMBUS
1.4b) SYSTEM: Yes.
```

Figure 1: An excerpt from a TRAINS 95 dialog (d95-162 from the ARPI tests)

Note that the user's utterances in the corpus are the hand-corrected output of the speech recognizer. In the raw output of the speech recognizer for utterance 1.1a, *and* is incorrectly translated as *in*. In addition, in utterance 1.2a, *Richmond* is translated into *reach noon* and in utterance 1.4a, *the engine* is translated into *begins in*. There are also several instances where *to go* is translated into *ago*. The outputs of the speech recognizer are displayed, so that users are aware of its unreliability. This is another reason why users keep their inputs simple. Utterance 1.3a does contain a PP attachment ambiguity; the user might be suggesting use of the engine that is in Charlotte, or performance of the *use*-action in Charlotte.

The 95 system's technique of always attaching to the NP is correct in this case, and for most of the inputs to the 95 system. The reason is that the transitive verbs typically used in these dialogs involve movement: *send*, *take*, *move*, and so on. These verbs are used in simple imperatives, and never in utterances such as those in figure 2. In addition, the grammar of the TRAINS 95 system is tailored to its domain, so as to eliminate many ambiguities that would be present in a more general setting. For example, many prepositions such as *to* are not allowed by the grammar to attach to

the NP.<sup>1</sup> The remaining ambiguous prepositions usually involve location: *about*, *near*, *in*, etc., and attach most often to the NP when it is the object of a movement verb.

The boxcar is at Corning.

We need to unload the boxcar at Elmira.

Figure 2: Samples of VP attachment

For an indication of disambiguation problems introduced by a more complex problem domain, we can look at the TRAINS 91-93 dialogs. This corpus was created between 1991 and 1993 from discussions between humans on transportation problems involving trains. Figure 3 shows an excerpt from such a dialog. We see that dealing with time constraints and with the added complexity of using engines to pick up boxcars and commodities to accomplish delivery goals leads to a less command-oriented dialog. In addition, since these dialogs involve two people, the speech recognition errors by the participants are very low, encouraging long and complex utterances.

```
M: _we_ have to ship a boxcar of oranges to Bath
   : by 8 AM
   : and it is now midnight
S: okay
M: okay all right so there are two boxcars at Bath
   and one at Dansville and there's
S: and there's
M: wait I've forgotten where the oranges are
   where are the oranges
S: the oranges are in the warehouse at Corning
M: okay so we need to get a boxcar to Corning
S: right
M: alright so why don't we take one of the ones from Bath
```

Figure 3: An excerpt from a TRAINS 91 dialog (d91-7.1)

We claim that this added complexity calls for more informed attachment decisions than are obtainable by a simple default method – despite the fact that default PP-attachment to the VP was observed to yield 88.7% accuracy in the 91-93 dialogs. There are two reasons for this. First, the probability of getting an utterance wrong because of an attachment error is higher in the 91-93 dialogs than in the 95 dialogs, since the average number of attachment ambiguities per utterance is much higher in the 91-93 dialogs than in the 95 dialogs. On average, a postmodifier attachment ambiguity appears in the 91-93 dialogs after about 43 words which is more frequent than the 75 word average of the 95 dialogs.

Second, the greater complexity of the 91-93 dialogs makes it much harder to “guess” the user’s intentions from utterance fragments, as a way of overcoming errors. In TRAINS 95, the options available to the user are sufficiently limited so that reasonable guesses about the actions a user is proposing can often be made from utterance fragments, even in the presence of multiple speech recognition and parsing errors. For instance, a PP of form *at city-name* can usually be assumed to

---

<sup>1</sup>NPs such as *the train to Corning* have not come up in the TRAINS dialogs.

- 2a) We need an engine for it.
- 2b) We met times for both deliveries.
- 3a) You have two things on the same train.
- 3b) I don't know if I can give any help on that.

Figure 4: Sample sentences from the TRAINS 91 and 93 dialogs

give the current location of an engine that is to be moved, while *to city-name* gives a destination for a proposed move. In the 91-93 dialogs, PP usage is much less predictable, both syntactically and semantically. Figure 4 shows some of the uses of *for* and *on* in TRAINS 91-93.

In sentence 2a, *for it* attaches to the VP headed by *need*, and semantically seems to supply something like a beneficiary (i.e., *for* can be read as something like “for the sake of”). In sentence 2b, *for both deliveries* seems to attach to the NP, *times*, and it would be a mistake to view this PP as supplying a beneficiary. Sentences 3a and 3b illustrate two very different uses of *on*. In 3a, the attachment of the PP, *on the train* is probably to the VP as a verb complement, and the meaning is locative. In 3b, the attachment of the PP, *on that* is probably to the NP, *any help*, and supplies the subject of discussion on which help has been requested, rather than a location. Clearly a simplistic attachment strategy that ignores the heads of the phrases involved in the alternative attachments would fail in such cases. Furthermore, it would be difficult to recover much information from the PPs considered in isolation.

## 2 Previous Work on PP Attachment Systems

Currently, the best PP attachment disambiguation techniques combining high accuracy with low computational cost involve recording phrase head patterns from corpora. Three teams using such an approach are Hindle and Rooth [HR93], Brill and Resnik [BR94], and Collins and Brooks [CB95]. Hindle and Rooth used a corpus of AP news stories to record PP attachment probabilities given the heads of the VP, NP, and PP involved. The corpus they used was not parsed in advance so the correct attachments were unknown. However, their idea was that co-occurrence statistics of phrase heads could themselves be used to make rather accurate guesses about correct attachments, and hence to obtain attachment probabilities for ambiguous cases. In particular, they noted that if a given preposition followed a given verb more often than it followed a given NP head (in the corpus as a whole), then in an ambiguous occurrence of form V NP PP, the most probable attachment is to the VP (and vice versa).<sup>2</sup> Treating the attachments obtained in this way as correct, they then ran a training phase in which they obtained the “correct” attachment probabilities for ambiguous cases. At the end of the training phase, Hindle and Rooth had a table indexed by the phrase heads and giving the probabilities of VP and NP attachment. Note that such a table reflects information about both ambiguous and unambiguous attachments in the corpus.

Brill and Resnik constructed a similar system but used parsed corpora and word classes and considered the objects of the PPs that they were attaching. Their major advance was considering patterns larger than bigrams. Unlike Hindle and Rooth’s approach, their approach was rule-based, and allowed attachment rules to specify any pattern of verb, preposition, NP head, and PP object head. A sample rule is shown in figure 5; it states that if the preposition is *in* and the head of the NP is in the word class *written-communication*, then attachment is made to the VP. The order of

---

<sup>2</sup>In a case such as *leave the engine at Elmira*, *at* is counted as following both *leave* and *engine*.

Attach to VP if P is IN and N1 is in class WRITTEN-COMMUNICATION.

Figure 5: A sample rule from Brill and Resnik's system

the rules is important since more than one may apply. Brill and Resnik created an ordered list of rules using a greedy search algorithm that at each step, adds the new rule to the list that gives the greatest increase in accuracy.

Collins and Brooks consider the same patterns as Brill and Resnik; these patterns are formed from the verb, NP head, preposition and PP object head around an ambiguous attachment. Instead of forming attachment rules, Collins and Brooks record attachment statistics on such patterns, and employ these in a backed-off manner. Specifically, when an ambiguous attachment is seen, Collins and Brooks first consider the full 4-gram around the attachment. If the particular pattern was not in the training corpus, then trigrams of elements around the attachment are considered.<sup>3</sup> If no trigrams matching the current situation were in the training corpus, Collins and Brooks consider bigrams. Scores for each attachment are computed from the frequency with which the patterns (4-gram, trigram, or bigram) occur with VP and NP attachments in the training corpus.

### 3 The KANKEI Disambiguation System

KANKEI<sup>4</sup> is a prototype disambiguation module for the TRAINS system. It resembles the systems of Collins and Brooks and Hindle and Rooth since they all record attachment statistics on information extracted from a corpus.<sup>5</sup> KANKEI is more akin to the system of Collins and Brooks since both consider patterns formed from four elements (verb, NP-head, preposition, and PP object head) around an ambiguous attachment. Both systems calculate scores for NP and VP attachments based on the frequencies with which these patterns appear in such attachments in a training corpus. However, unlike Collins and Brooks' backed-off model, KANKEI uses a weighted sum of trigrams and bigrams if it cannot find the appropriate 4-gram in its training corpus. Currently these weights are set to unity, but in future work they will be adjusted to optimize performance. In addition, future work will consider the possibility of always using a weighted sum of the 4-gram, trigrams, and bigrams around an attachment.

Like Brill and Resnik and Collins and Brooks, we used hand-parsed corpora to avoid the parse errors that occur in automatically parsed corpora. The first hand-parsed corpus used in these experiments consisted of 2548 instances from the TRAINS 91-93 dialogs of PP or adverb postmodifiers that can attach to either VPs or NPs. Many of these cases were unambiguous; there was no NP following the VP, or the NP did not follow a verb. Only 1585 instances contained ambiguous constructions. The second hand-parsed corpus consisted of 640 instances from the TRAINS 95 dialogs of PP and

---

<sup>3</sup>Collins and Brooks showed that patterns without prepositions have poor predictive value and thus did not use trigrams or bigrams without prepositions.

<sup>4</sup>From the Japanese word, *kankei*, meaning "relation."

<sup>5</sup>This paper is not advocating such an approach over the rule-based approach of Brill and Resnik, if the goal is simply to get the best attachments possible based on patterns of phrasal heads in a parsed corpus. However, for future integration with other syntactic, semantic and pragmatic clues to disambiguation, we preferred to frame the problem as a calculation of attachment scores.

verb NP-head (preposition obj-head | adverb1 adverb2)

Figure 6: Format of an attachment pattern

adverb postmodifiers that can attach to either VPs or NPs.<sup>6</sup> 275 of these instances contained both an NP and VP to which the postmodifier could attach.

### 3.1 Patterns of phrasal heads

As shown in figure 6, the patterns used here allow for combinations of a verb, object head, and the preposition and head noun of a PP, or an adverb (possibly preceded by another adverb).<sup>7</sup> Note that most items in this specification are optional. Only the last preposition (with or without obj-head), or an adverb, must be present, but it must be preceded by at least one other item. Even the verb may be missing, since there are utterances consisting only of an NP. The singular forms of nouns and the base forms of verbs are used. These patterns (with hyphens separating the items) form keys to two hash tables; one records attachments to NPs while the other records attachments to VPs. Numbers are stored under these keys to record how often such a pattern was seen in a VP or NP attachment. Sentence 4 instantiates the longest possible pattern, a 4-gram that here consists of *need*, *orange*, *in*, and *Elmira*. Sentence 4 illustrates why it is helpful to include the object of the ambiguous PP in the pattern: if *Elmira* were replaced with *two hours* then the attachment would be different.

4) I need the oranges in Elmira.

### 3.2 Dealing with a high-dimensional space

In the two TRAINS corpora, most pattern instances do not have the maximal number of constituents. However, even if we are primarily recording trigrams or bigrams, our corpora are much too small to provide an adequate sample of these n-gram patterns. While searching for attachment statistics for sentence 4, KANKEI will check its hash tables for the key *need-orange-in-Elmira*. If KANKEI relied entirely on full patterns, then if the pattern had not been seen, KANKEI would have to randomly guess the attachment. Such a technique will be referred to as *full matching*.

Normally KANKEI will do *partial matching*, i.e., if it cannot find a pattern such as *need-orange-in-Elmira*, it will look for smaller partial patterns which here would be: *need-in*, *orange-in*, *orange-in-Elmira*, *need-in-Elmira*, and *need-orange-in*. More generally, the partial patterns are:

- verb-preposition/adverb
- NPhead-preposition/adverb
- verb-NPhead-preposition/adverb

---

<sup>6</sup>In both of these corpora, postmodifiers such as *at five o'clock* sometimes appear at the beginning of the utterance. Such examples are used in this study as unambiguous VP attachments.

<sup>7</sup>Apart from allowing for adverb attachment, these patterns are similar to Brill and Resnik's and Collins and Brooks' patterns. Examples of trailing adverb pairs are *first off* and *right now*. Actually, the "adverb1 adverb2" slots are used rather permissively, also allowing for idiomatic pairs involving prepositions/particles, adjectives, and nouns, as in *at most*, *by way*, etc.

\*Commodity\*: orange(s), juice, oj, banana(s)  
\*City\*: city, town, (various city, state, and country names)  
\*Train\*: train, engine, boxcar, tanker, car, e1, etc.

Figure 7: Word classes used in KANKEI

- verb-preposition/adverb-PPobjectHead/adverb2
- NPhead-preposition/adverb-PPobjectHead/adverb2

When KANKEI examines these partial patterns, it totals the frequencies with which they were seen in NP and VP attachments. The attachment with the greatest support is chosen. Instead of counting partial patterns equally, we would like to give reduced credit to shorter patterns such as *need-in*. Also, some elements such as verbs may have greater predictive power. Such refinements will be pursued in future work.

The statistics used by KANKEI for partial or full matching can be obtained in various ways. One is to use the same kinds of full and partial pattern matching in training as are used in disambiguation. This is called *comprehensive training*. Another method, called *raw training*, is to record only full patterns for ambiguous and unambiguous attachments in the corpus. (Note that full patterns can be as small as bigrams, such as when an adverb follows an NP acting as a subject.) Although raw training only collects a subset of the data collected by comprehensive training, it still gives KANKEI some flexibility when disambiguating phrases. If the full pattern of an ambiguity has not been seen, KANKEI can test whether a partial pattern of this ambiguous attachment occurred as an unambiguous attachment in the training corpus.

Another technique we have implemented to deal with the high dimensionality is to replace words with word classes, as Brill and Resnik did. For example, consider sentences 5 and 6. We would like to train on sentence 5 and then be able to predict sentence 6 by use of the generalization that *Elmira* and *Corning* are both cities. Thus, when constructing keys during training or testing, experiments were performed in which words were replaced with their word classes.

- 5) Take the train in Elmira.
- 6) Take the train in Corning.

Constructing a good set of word classes is a topic outside the focus of this paper. Figure 7 shows the crude set of classes used in evaluation of this system.

## 4 Examples of Training and Testing

The first step in setting up KANKEI was to collect a corpus of training utterances containing PPs and adverbs which can attach to either a VP or an NP in the utterance. Figure 8 lists three such sentences from the 91-93 dialogs. The parsed version of these sentences is shown in figure 9. Note that the time, “9 a.m.,” in sentence 7 is replaced by the symbol, \*TIME\*. Even when not using word classes, this replacement is used in order to avoid having to deal with numbers.

- 7) That will get us there by 9 a.m.
- 8) There's boxcars at Bath, Dansville, and Elmira.
- 9) The oranges are in the warehouse at Corning.

Figure 8: Sentences from the 91-93 dialogs containing PPs that can attach to NPs or VPs

```

7p)(S (DP THAT) (VP WILL (VP GET (DP US) (DP THERE) (PP BY *TIME*))))
8p)(S (DP THERE) (VP BE (NP BOXCAR) (PP AT (DP BATH))))
9p)(S (DP ORANGE) (VP BE (PP IN (DP WAREHOUSE (PP AT (DP CORNING))))))

```

Figure 9: Parsed sentences from the 91-93 dialogs containing PPs that can attach to NPs or VPs

The VP attachment hash table shown in table 1 contains the partial patterns of heads surrounding VP attachments in these three sentences. Note that these examples contain partial patterns with word classes. KANKEI can be set so as to record full patterns only and so as not to use word classes. Table 1 shows the partial patterns from *by 9 a.m* attaching to the VP headed by *get*, *at Bath* attaching to the VP headed by *be*, and *in warehouse* attaching to the VP headed by *be*. The NP attachment hash table shown in table 2 contains the partial patterns involving the attachment of *at Corning* to the NP headed by *warehouse*.

After patterns have been recorded, the parser has NP and VP attachment hash tables reflecting the attachments in the training corpus. When KANKEI encounters a PP/adverb attachment ambiguity in testing, it looks up in the hash tables the full pattern of verb, preposition, and object heads involved in the ambiguity. If the pattern has not been seen and KANKEI is using partial matching, then the partial patterns of verb, preposition, and object heads are looked up in the NP and VP attachment hash tables.

To show how this works, we consider a particular training run and subsequent performance on some sample sentences. KANKEI was trained on 75% of the 91-93 dialogs using partial patterns and word classes, and was then tested on three sample sentences (see figure 10) from the remaining 25% of the 91-93 dialogs. Tables 3 and 4 show the relevant portions of the VP and NP attachment tables used for these examples. These examples were chosen because the full patterns of most of them were not seen in training.

For sentence 10, the partial pattern, HAVE-IN was found in both the NP and VP attachment tables. HAVE-\*COMMODITY\*-IN was found only in the VP attachment table. The total evidence for VP attachment was 7 partial patterns seen in training while the evidence for NP attachment was 2 partial patterns. Thus, the VP attachment is correctly chosen here. In sentence 11, the partial pattern PICK-\*NUM\*-AT<sup>8</sup> had been seen three times in VP attachments while the partial pattern \*NUM\*-AT was seen four times in NP attachments. NP attachment is incorrectly chosen here (perhaps due to the equal treatment of the bigram and the trigram).

In sentence 12, we correctly attach *at noon* to the VP because the full pattern of GET-\*CITY\*-AT-NOON was seen in a VP attachment. We do not immediately know whether to attach *on the previous plan* to *noon* or the VP headed by *get* because the full pattern is not present. Based on the partial patterns, VP attachment is correctly chosen because in the VP hash table, GET-ON occurs with frequency 7 and GET-ON-PLAN occurs with frequency 1.

---

<sup>8</sup>NPs such as *the one* or *the three* are always represented by \*NUM\* even when word classes are not used. This technique allows us to avoid dealing with numbers.



| Key                 | Value |
|---------------------|-------|
| BE-AT               | 1     |
| BOXCAR-AT           | 1     |
| BE-BOXCAR-AT        | 1     |
| BE-AT-BATH          | 1     |
| BOXCAR-AT-BATH      | 1     |
| BE-BOXCAR-AT-BATH   | 1     |
| BE-IN               | 1     |
| BE-IN-WAREHOUSE     | 1     |
| GET-BY              | 1     |
| THERE-BY            | 1     |
| GET-BY-*TIME*       | 1     |
| GET-THERE-BY        | 1     |
| THERE-BY-*TIME*     | 1     |
| GET-THERE-BY-*TIME* | 1     |

Table 1: VP attachment hash table

| Key                     | Value |
|-------------------------|-------|
| BE-AT                   | 1     |
| WAREHOUSE-AT            | 1     |
| BE-WAREHOUSE-AT         | 1     |
| BE-AT-CORNING           | 1     |
| WAREHOUSE-AT-CORNING    | 1     |
| BE-WAREHOUSE-AT-CORNING | 1     |

Table 2: NP attachment hash table

- 10) The boxcar that has oranges in it.
- 11) Why don't we pick the one up at Dansville.
- 12) The orange juice was gonna get to Avon at noon on the previous plan.

Figure 10: Sample sentences from the 91-93 dialogs containing PPs that can attach to NPs or VPs

| Key                 | Value |
|---------------------|-------|
| *NUM*-AT            | 4     |
| GET-*CITY*-AT-NOON  | 5     |
| GET-ON              | 7     |
| GET-ON-PLAN         | 1     |
| HAVE-IN             | 4     |
| HAVE-*COMMODITY*-IN | 3     |
| PICK-*NUM*-AT       | 3     |

Table 3: VP attachment hash table

| Key      | Value |
|----------|-------|
| *NUM*-AT | 4     |
| HAVE-IN  | 2     |

Table 4: NP attachment hash table

| Accuracy | Word Classes | Raw Training | Partial Matching | Default Attachment |
|----------|--------------|--------------|------------------|--------------------|
| 90.9%    | Yes          | Yes          | Yes              | VP                 |
| 87.6%    | Yes          | Yes          | Yes              | NP                 |
| 86.9%    | Yes          | No           | No               | NP                 |
| 85.5%    | Yes          | Yes          | No               | NP                 |

Table 5: Results of training with the 91-93 dialogs and testing on the 95 dialogs

## 5 Experiments

The test examples in the previous section used partial patterns and word classes and involved training and testing on the 91-93 dialogs. The following experiments involved determining the effect of using partial patterns and word classes on making correct attachments. An additional parameter is the choice of corpora to use for training and testing. Since there are two distinct corpora, we can see how well each predicts itself and how well they predict each other.

To test how well they predict themselves, the 91-93 corpus was divided into four parts and the 95 corpus into two parts (because of its small size). To test, for example, how well the 91-93 corpus predicted itself, KANKEI was trained on the first three sections and tested on the fourth. These steps were repeated by training on the last three sections and testing on the first. This continued until all four sections were tested.

Several parameters determine how patterns are used during disambiguation, and the experiments were designed to test for any performance differences for alternative settings of these parameters. One parameter chooses between comprehensive and raw training. Another chooses between full matching and partial matching. A third decides whether the default attachment is to the NP or to the VP, in cases where the (weighted) evidence does not favor either attachment.

The test results supported the notion that the 95 and 91-93 dialogs are very different. In the 95 dialogs, NP attachments occur 73.5% of the time for PPs following a verb and NP, while in the 91-93 dialogs, VP attachments occur more frequently (88.7% of the time). A gratifying result is that the 91-93 dialogs predicted attachments in the 95 test data with reasonable accuracy (a success rate of 90.9%). Thus, this disambiguation scheme seems capable of making generalizations that are independent of the corpus from which they were drawn. To understand the role of word classes and partial patterns in these results, we need to examine the data more closely. Table 5 lists the top four accuracies obtained by using the 91-93 dialogs to predict the 95 dialogs. Next to each accuracy are the associated parameter settings for that test run.

The reason word classes were so important was that the 95 data includes over 40 city names of which the 91-93 data only includes five. The difference between 90.9% accuracy and 86.9% accuracy is 11 attachments. This suggests that partial patterns are important in making generalizations that

| Accuracy | Word Classes | Raw Training | Partial Matching | Default Attachment |
|----------|--------------|--------------|------------------|--------------------|
| 92.4%    | Yes          | No           | Yes              | NP                 |
| 92.0%    | Yes          | No           | Yes              | VP                 |
| 92.0%    | No           | No           | Yes              | VP                 |
| 91.6%    | Yes          | Yes          | Yes              | VP                 |
| 90.9%    | No           | No           | Yes              | NP                 |

Table 6: Results of training and testing on the 95 dialogs

| Accuracy | Word Classes | Raw Training | Partial Matching | Default Attachment |
|----------|--------------|--------------|------------------|--------------------|
| 92.2%    | Yes          | No           | Yes              | VP                 |
| 91.3%    | No           | No           | Yes              | VP                 |
| 91.0%    | Yes          | Yes          | No               | VP                 |
| 91.0%    | Yes          | No           | No               | VP                 |
| 90.3%    | No           | Yes          | No               | VP                 |
| 90.3%    | No           | No           | No               | VP                 |

Table 7: Results of training and testing on the 91-93 dialogs

apply across the two different domains. However, more data is necessary before definite conclusions can be drawn.

When training on the 95 dialogs to test on the 95 dialogs, most of the parameter settings worked better than always guessing NP attachment. Table 6 lists the top five results. The difference between 92.4% accuracy and 91.6% is only two attachments. However, when partial matching is not used, a drop in 16 attachments from 92.4% accuracy is seen. This drop suggests the importance of partial matching.

Since the 91-93 data consisted of more complex dialogs, it is not surprising that the best attachment accuracy KANKEI achieved was somewhat lower, 92.2%. This accuracy was obtained by using 91-93 data for training. Using the 95 data for training was at best as good as always attaching to the VP. Table 7 contains the six parameter settings associated with training and testing on the 91-93 data that achieved an accuracy better than 90.0%. The difference between 92.2% accuracy and 91.3% is 15 attachments which suggests the importance of word classes. The difference between 92.2% accuracy and 91.0% is 19 attachments, suggesting the importance of partial matching and comprehensive training. The best result with no word classes, no comprehensive training, and no partial matching is 30 attachments worse than the overall best. It appears that generalization techniques are necessary to achieve the best accuracy possible.

## 6 Conclusions

One important result was the observation that a change in the domain made a large difference in the PP/adverb attachment characteristics. The 95 dialogs had 73.5% NP attachments while the 91-93 dialogs had 88.7% VP attachments. This result is strong evidence against a universal default attachment for English. This evidence is interesting as both sets of dialogs are about planning

| Training Data | Test Data  | Best Results |
|---------------|------------|--------------|
| 91-93 data    | 91-93 data | 88.9%        |
| 91-93 data    | 95 data    | 93.1%        |
| 95 data       | 95 data    | 91.6%        |
| 95 data       | 91-93 data | 86.9%        |

Table 8: Results of using bigram statistics

train routes. The domain of the 91-93 dialogs contains more objects and constraints to discuss but perhaps the major difference is that the 91-93 dialogs are between people while the 95 dialogs consist of human-computer conversations.

A second important result was how the 91-93 dialogs were able to predict attachments in the 95 dialogs reasonably well (90.9% accuracy versus 92.4% when using 95 dialogs for training). Since the corpora are very different, KANKEI was evidently robust enough to make generalizations that held over the two domains. Partial patterns and word classes were the mechanisms for this generalization. Most of the best parameter settings used word classes and/or some form of partial patterns.

These results are better than those of Hindle and Rooth but they used a more general corpus which probably lowers their accuracy considerably. To make comparisons, KANKEI was modified to act like the system of Hindle and Rooth (so as to work only with bigrams). For the most part, the results were worse. However, if word classes were used then training on the 91-93 data and testing on 95 data gave better results (93.1%) than the experiments including trigram and 4-gram patterns (90.9%, a difference of six attachments). However, limiting KANKEI to bigrams gives worse results overall (see table 8). The anomalous result suggests that proper weighting of partial patterns will be helpful since in this particular case, the bigram patterns alone provided better accuracy than the combined evidence of all the patterns. On the other hand, the uneven results may be due simply to insufficient data. Some of the results in table 8 were obtained without word classes so there is no straightforward evidence supporting word classes in this experiment.

Since for the most part the accuracies obtained with bigram data alone are lower than those of the main experiments, it seems that the extra information gained by simultaneously considering the verb, NP head, and preposition and PP object head (or adverbs) gives us an advantage. Using word classes and/or partial patterns, generalizations that hold across corpora of different domains can be made. Future improvements should be possible with more careful choices of word classes and weightings of partial patterns.<sup>9</sup> As the TRAINS system starts to handle some of the 91-93 dialogs, KANKEI will become increasingly important.

## 7 Future Work

One of the most important unaccomplished tasks of this work is integrating KANKEI into the TRAINS parser. One problem with integrating disambiguation systems into parsers is that of deciding when to disambiguate. Waiting until the end of the sentence before performing disambiguation provides more evidence with which to make a decision. However, if parsers wait until the end of

---

<sup>9</sup>Of course there is a limit to the accuracy simple co-occurrence statistics can provide. Ultimately, a “smarter” disambiguation module is needed to boost performance above this limit.

the sentence to disambiguate, they have the overhead of computing all the alternative parses. The techniques in this paper can be used before sentence completion, as soon as a PP is encountered; in a left-to-right parse, the required phrase heads will already be available.

The current TRAINS parser does not discard dispreferred parses. Instead, it assigns them lower probabilities and works on higher probability parse trees first. KANKEI could be incorporated in modified form into the parser. The number of patterns supporting an attachment could be used to adjust the corresponding probabilities, rather than being used to make an immediate decision.

Later, when different disambiguation systems (e.g., ones sensitive to semantic and pragmatic factors) are added to the TRAINS system, they could be used similarly to adjust the probabilities of constituents. Such an approach is recommended by Schubert in [Sch86] and allows various disambiguation systems to act independently. The idea is that different disambiguation systems would “vote” for various constituents by imparting “activation potentials” to them. (This is similar to boosting their probabilities.) These potentials would be propagated upward in phrase structure trees to the root, and low-potential trees would be discarded. This pruning can be done on-line, before utterance completion. When an utterance is complete, the highest-potential tree would be chosen.

Another important element in future work on this system is to develop an algorithm to assign weights to partial patterns and test how such a system compares to the rule learning system of Brill and Resnik. Alshawi and Carter [AC94] address a related problem, weighting scores from different disambiguation systems to obtain a single rating for a parse tree. They achieved good results using a hill climbing technique to explore the space of possible weights. Thus, future work will include trying such a method for training weights in KANKEI. Another possible technique for combining evidence is the maximum-entropy technique of Wu described in [Wu93]. Additional future improvements to KANKEI will include taking into account the definiteness of the NP in the patterns, VP ... NP PP and NP ... NP adverb(s), as suggested by Spivey-Knowlton and Sedivy in [SKS95]. In addition, further research is necessary in order to find a better set of word classes for the TRAINS domains as well as more general settings.

In addition to making improvements to KANKEI, we need to investigate the differences between these head-based patterns and the traditional patterns of adjacent words used by context-sensitive statistics. Specifically, we need to investigate what information head-based patterns may be missing and whether they can be applied to other forms of structural ambiguity.

A related topic is the relation between making attachment decisions and lexical disambiguation. Sentences 2a and b repeated here as 13a and b give an indication of this relationship. *for it* seems to supply something like a beneficiary in contrast to *for both deliveries* which specifies to what *times* refers. One would not count *for* as being the same lexical item in both sentences. Further investigation is needed into how different meanings give different attachments and how different attachments provide evidence about different lexical meanings.

13a) We need an engine for it.

13b) We met times for both deliveries.

## References

- [AC94] H. Alshawi and D. Carter. Training and scaling preference functions. *Computational Linguistics*, 20(1):635–648, 1994.
- [AFMR95] J.F. Allen, G. Ferguson, B. Miller, and E. Ringger. Spoken dialogue and interactive planning. In *Proc. of the ARPA Spoken Language Technology Workshop*, Austin, TX, 1995.
- [BR94] E. Brill and P. Resnik. A rule-based approach to prepositional phrase attachment disambiguation. In *15th International Conference on Computational Linguistics (COLING-94)*, Kyoto, Japan, 1994.
- [CB95] M. Collins and J. Brooks. Prepositional phrase attachment through a backed-off model. In *Proc. of the Third Workshop on Very Large Corpora*, pages 27–38, Boston, MA, June 1995.
- [HR93] D. Hindle and M. Rooth. Structural ambiguity and lexical relations. *Computational Linguistics*, 19(1):103–120, 1993.
- [Sch86] L.K. Schubert. Are there preference trade-offs in attachment decisions. In *Proc. AAAI-86*, pages 601–605, Philadelphia, PA, 1986.
- [SKS95] Michael Spivey-Knowlton and Julie C. Sedivy. Resolving attachment ambiguities with multiple constraints. *Cognition*, 55:227–267, 1995.
- [Wu93] D. Wu. Estimating probability distributions over hypotheses with variable unification. In *Proc. of the Eleventh National Conference on Artificial Intelligence*, pages 790–795, Washington D.C., 1993.