Evaluating Automated and Manual Acquisition of Anaphora Resolution Strategies

Chinatsu Aone and **Scott William Bennett**
Systems Research and Applications Corporation (SRA)
2000 15th Street North
Arlington, VA 22201
aonec@sra.com, bennett@sra.com

Abstract

We describe one approach to build an automatically trainable anaphora resolution system. In this approach, we use Japanese newspaper articles tagged with discourse information as training examples for a machine learning algorithm which employs the C4.5 decision tree algorithm by Quinlan (Quinlan, 1993). Then, we evaluate and compare the results of several variants of the machine learning-based approach with those of our existing anaphora resolution system which uses manually-designed knowledge sources. Finally, we compare our algorithms with existing theories of anaphora, in particular, Japanese zero pronouns.

1 Introduction

Anaphora resolution is an important but still difficult problem for various large-scale natural language processing (NLP) applications, such as information extraction and machine translation. Thus far, no theories of anaphora have been tested on an empirical basis, and therefore there is no answer to the "best" anaphora resolution algorithm.[1] Moreover, an anaphora resolution system within an NLP system for real applications must handle:

- degraded or missing input (no NLP system has complete lexicons, grammars, or semantic knowledge and outputs perfect results), and
- different anaphoric phenomena in different domains, languages, and applications.

Thus, even if there exists a perfect theory, it might not work well with noisy input, or it would not cover all the anaphoric phenomena.

These requirements have motivated us to develop robust, extensible, and trainable anaphora resolution systems. Previously (Aone and McKee, 1993), we reported our data-driven multilingual anaphora resolution system, which is robust, extensible, and manually trainable. It uses discourse knowledge sources (KS's) which are manually selected and ordered. (Henceforth, we call the system the Manually-Designed Resolver, or MDR.) We wanted to develop, however, truly automatically trainable systems, hoping to improve resolution performance and reduce the overhead of manually constructing and arranging such discourse data.

In this paper, we first describe one approach we are taking to build an automatically trainable anaphora resolution system. In this approach, we tag corpora with discourse information, and use them as training examples for a machine learning algorithm. (Henceforth, we call the system the Machine Learning-based Resolver, or MLR.) Specifically, we have tagged Japanese newspaper articles about joint ventures and used the C4.5 decision tree algorithm by Quinlan (Quinlan, 1993). Then, we evaluate and compare the results of the MLR with those produced by the MDR. Finally, we compare our algorithms with existing theories of anaphora, in particular, Japanese zero pronouns.

2 Applying a Machine Learning Technique to Anaphora Resolution

In this section, we first discuss corpora which we created for training and testing. Then, we describe the learning approach chosen, and discuss training features and training methods that we employed for our current experiments.

2.1 Training and Test Corpora

In order to both train and evaluate an anaphora resolution system, we have been developing corpora which are tagged with discourse information. The tagging has been done using a GUI-based tool called the Discourse Tagging Tool (DTTool) according to "The Discourse Tagging Guidelines" we

[1] Walker (Walker, 1989) compares Brennan, Friedman and Pollard's centering approach (Brennan et al., 1987) with Hobbs' algorithm (Hobbs, 1976) on a theoretical basis.

have developed.[2] The tool allows a user to link an anaphor with its antecedent and specify the type of the anaphor (e.g. pronouns, definite NP's, etc.). The tagged result can be written out to an SGML-marked file, as shown in Figure 1.

For our experiments, we have used a discourse-tagged corpus which consists of Japanese newspaper articles about joint ventures. The tool lets a user define types of anaphora as necessary. The anaphoric types used to tag this corpus are shown in Table 1.

NAME anaphora are tagged when proper names are used anaphorically. For example, in Figure 1, "Yamaichi (ID=3)" and "Sony-Prudential (ID=5)" referring back to "Yamaichi Shouken (ID=4)" (Yamaichi Securities) and "Sony-Prudential Seimeihoken (ID=6)" (Sony-Prudential Life Insurance) respectively are NAME anaphora. NAME anaphora in Japanese are different from those in English in that any combination of *characters* in an antecedent can be NAME anaphora as long as the character order is preserved (e.g. "abe" can be an anaphor of "abcde").

Japanese definite NPs (i.e. DNP anaphora) are those prefixed by "dou" (literally meaning "the same"), "ryou" (literally meaning "the two"), and deictic determiners like "kono"(this) and "sono" (that). For example, "dou-sha" is equivalent to "the company", and "ryou-koku" to "the two countries". The DNP anaphora with "dou" and "ryou" prefixes are characteristic of written, but not spoken, Japanese texts.

Unlike English, Japanese has so-called zero pronouns, which are not explicit in the text. In these cases, the DTTool lets the user insert a "Z" marker just before the main predicate of the zero pronoun to indicate the existence of the anaphor. We made distinction between QZPRO and ZPRO when tagging zero pronouns. QZPRO ("quasi-zero pronoun") is chosen when a sentence has multiple clauses (subordinate or coordinate), and the zero pronouns in these clauses refer back to the subject of the initial clause in the *same* sentence, as shown in Figure 2.

The anaphoric types are sub-divided according to more semantic criteria such as organizations, people, locations, etc. This is because the current application of our multilingual NLP system is information extraction (Aone et al., 1993), i.e. extracting from texts information about which organizations are forming joint ventures with whom. Thus, resolving certain anaphora (e.g. various ways to refer back to organizations) affects the task performance more than others, as we previously reported (Aone, 1994). Our goal is to customize and evaluate anaphora resolution systems according to the types of anaphora when necessary.

2.2 Learning Method

While several inductive learning approaches could have been taken for construction of the trainable anaphoric resolution system, we found it useful to be able to observe the resulting classifier in the form of a decision tree. The tree and the features used could most easily be compared to existing theories. Therefore, our initial approach has been to employ Quinlan's C4.5 algorithm at the heart of our classification approach. We discuss the features used for learning below and go on to discuss the training methods and how the resulting tree is used in our anaphora resolution algorithm.

2.3 Training Features

In our current machine learning experiments, we have taken an approach where we train a decision tree by feeding *feature vectors* for pairs of an anaphor and its possible antecedent. Currently we use 66 features, and they include *lexical* (e.g. category), *syntactic* (e.g. grammatical role), *semantic* (e.g. semantic class), and *positional* (e.g. distance between anaphor and antecedent) features. Those features can be either *unary* features (i.e. features of either an anaphor or an antecedent such as syntactic number values) or *binary* features (i.e. features concerning relations between the pairs such as the positional relation between an anaphor and an antecedent.) We started with the features used by the MDR, generalized them, and added new features. The features that we employed are common across domains and languages though the feature values may change in different domains or languages. Example of training features are shown in Table 2.

The feature values are obtained automatically by processing a set of texts with our NLP system, which performs lexical, syntactic and semantic analysis and then creates *discourse markers* (Kamp, 1981) for each NP and S.[3] Since discourse markers store the output of lexical, syntactic and semantic processing, the feature vectors are automatically calculated from them. Because the system output is not always perfect (especially given the complex newspaper articles), however, there is some noise in feature values.

2.4 Training Methods

We have employed different training methods using three parameters: anaphoric chains, anaphoric type identification, and confidence factors.

The anaphoric chain parameter is used in selecting training examples. When this parameter is *on*, we select a set of *positive* training examples and a set of *negative* training examples for each anaphor in a text in the following way:

[2] Our work on the DTTool and tagged corpora was reported in a recent paper (Aone and Bennett, 1994).

[3] Existence of zero pronouns in sentences is detected by the syntax module, and discourse markers are created for them.

```
<COREF ID="1"><COREF ID="4">山一証券</COREF>と<COREF ID="6">ソニー・プルデンシャル生命保険（平井竜明社長、本社・東京）
</COREF></COREF>は、顧客の開拓、情報提供などの分野で業務提携することにし、二日覚書に
<COREF ID="0" TYPE="QZPRO-ORG" REF="1"></COREF>調印した。四月中旬から<COREF ID="2" TYPE="ZPRO-ORG" REF="1"></COREF>実施
する。<COREF ID="3" TYPE="NAME-ORG" REF="4">山一</COREF>が<COREF ID="8">中小企業の節税などに役立つ財産管理情報システム
</COREF>を<COREF ID="5" TYPE="NAME-ORG" REF="6">ソニー・プルデンシャル</COREF>に提供、<COREF ID="7" TYPE="DNP" REF="8">
このシステム</COREF>を<COREF ID="9" TYPE="ZPRO-ORG" REF="5"></COREF>使って<COREF ID="10" TYPE="QZPRO-ORG" REF="5">
</COREF>獲得した顧客の証券運用を<COREF ID="11" TYPE="NAME-ORG" REF="3">山一</COREF>が担当する。
```

Figure 1: Text Tagged with Discourse Information using SGML

Table 1: Summary of Anaphoric Types

Tags	Meaning
DNP	Definite NP
DNP-F	Definite NP whose referent is a facility
DNP-L	Definite NP whose referent is a location
DNP-ORG	Definite NP whose referent is an organization
DNP-P	Definite NP whose referent is a person
DNP-T	Definite NP whose referent is time
DNP-BOTH	Definite NP whose referent is two entities
DNP-BOTH-ORG	Definite NP whose referent is two organization entities
DNP-BOTH-L	Definite NP whose referent is two location entities
DNP-BOTH-P	Definite NP whose referent is two person entities
REFLEXIVE	Reflexive expressions (e.g. "jisha")
NAME	Proper name
NAME-F	Proper name for facility
NAME-L	Proper name for location
NAME-ORG	Proper name for organization
NAME-P	Proper name for person
DPRO	Deictic pronoun (this, these)
LOCI	Locational indexical (here, there)
TIMEI	Time indexical (now, then, later)
QZPRO	Quasi-zero pronoun
QZPRO-ORG	Quasi-zero pronoun whose referent is an organization
QZPRO-P	Quasi-zero pronoun whose referent is a person
ZPRO	Zero pronoun
ZPRO-IMP	Zero pronoun in an impersonal construction
ZPRO-ORG	Zero pronoun whose referent is an organization
ZPRO-P	Zero pronoun whose referent is a person
JDEL	Dou-ellipsis

SONY-wa	RCA-to	teikeishi,	VCR-wo	QZPRO
Sony-subj	*RCA-with*	*joint venture*	*VCR-obj*	*(it)*
kaihatsusuru	to	QZPRO	happyoushita	
develop	*that*	*(it)*	*announced*	

"(SONY) announced that SONY will form a joint venture with RCA and (it) will develop VCR's."

Figure 2: QZPRO Example

Table 2: Examples of Training Features

	Unary feature	Binary feature
Lexical	category	matching-category
Syntactic	topicalized	matching-topicalized
Semantic	semantic-class	subsuming-semantic-class
Positional		antecedent-precedes-anaphor

Positive training examples are those anaphor-antecedent pairs whose anaphor is directly linked to its antecedent in the tagged corpus and also whose anaphor is paired with one of the antecedents on the *anaphoric chain*, i.e. the transitive closure between the anaphor and the first mention of the antecedent. For example, if B refers to A and C refers to B, C-A is a positive training example as well as B-A and C-B.

Negative training examples are chosen by pairing an anaphor with all the possible antecedents in a text except for those on the transitive closure described above. Thus, if there are possible antecedents in the text which are not in the C-B-A transitive closure, say D, C-D and B-D are negative training examples.

When the anaphoric chain parameter is *off*, only those anaphor-antecedent pairs whose anaphora are directly linked to their antecedents in the corpus are considered as positive examples. Because of the way in which the corpus was tagged (according to our tagging guidelines), an anaphor is linked to the most recent antecedent, except for a zero pronoun, which is linked to its most recent *overt* antecedent. In other words, a zero pronoun is never linked to another zero pronoun.

The anaphoric type identification parameter is utilized in training decision trees. With this parameter *on*, a decision tree is trained to answer "no" when a pair of an anaphor and a possible antecedent are not co-referential, or answer the anaphoric type when they are co-referential. If the parameter is *off*, a binary decision tree is trained to answer just "yes" or "no" and does not have to answer the types of anaphora.

The confidence factor parameter (0-100) is used in pruning decision trees. With a higher confidence factor, less pruning of the tree is performed, and thus it tends to overfit the training examples. With a lower confidence factor, more pruning is performed, resulting in a smaller, more generalized tree. We used confidence factors of 25, 50, 75 and 100%.

The anaphoric chain parameter described above was employed because an anaphor may have more than one "correct" antecedent, in which case there is no absolute answer as to whether one antecedent is better than the others. The decision tree approach we have taken may thus predict more than one antecedent to pair with a given anaphor. Currently, confidence values returned from the decision tree are employed when it is desired that a single antecedent be selected for a given anaphor. We are experimenting with techniques to break ties in confidence values from the tree. One approach is to use a particular bias, say, in preferring the antecedent closest to the anaphor among those with the highest confidence (as in the results reported here). Although use of the confidence values from the tree works well in practice, these values were only intended as a heuristic for pruning in Quinlan's C4.5. We have plans to use cross-validation across the training set as a method of determining error-rates by which to prefer one predicted antecedent over another.

Another approach is to use a hybrid method where a preference-trained decision tree is brought in to supplement the decision process. Preference-trained trees, like that discussed in Connolly *et al.* (Connolly et al., 1994), are trained by presenting the learning algorithm with examples of when one anaphor-antecedent pair should be preferred over another. Despite the fact that such trees are learning preferences, they may not produce sufficient preferences to permit selection of a single best anaphor-antecedent combination (see the "Related Work" section below).

3 Testing

In this section, we first discuss how we configured and developed the MLRs and the MDR for testing. Next, we describe the scoring methods used, and then the testing results of the MLRs and the MDR. In this paper, we report the results of the four types of anaphora, namely NAME-ORG, QZPRO-ORG, DNP-ORG, and ZPRO-ORG, since they are the majority of the anaphora appearing in the texts and most important for the current domain (i.e. joint ventures) and application (i.e. information extraction).

3.1 Testing the MLRs

To build MLRs, we first trained decision trees with 1971 anaphora[4] (of which 929 were NAME-ORG; 546 QZPRO-ORG; 87 DNP-ORG; 282 ZPRO-ORG) in 295 training texts. The six MLRs using decision trees with different parameter combinations are described in Table 3.

Then, we trained decision trees in the MLR-2 configuration with varied numbers of training texts, namely 50, 100, 150, 200 and 250 texts. This is done to find out the minimum number of training texts to achieve the optimal performance.

3.2 Testing the MDR

The same training texts used by the MLRs served as development data for the MDR. Because the NLP system is used for extracting information about joint ventures, the MDR was configured to handle only the crucial subset of anaphoric types for this experiment, namely all the name anaphora and zero pronouns and the definite NPs referring to organizations (i.e. DNP-ORG). The MDR applies different sets of *generators*, *filters* and *orderers* to resolve different anaphoric types (Aone and McKee, 1993). A generator generates a set of possible antecedent hypotheses for each anaphor, while a filter eliminates

[4] In both training and testing, we did not include anaphora which refer to multiple discontinuous antecedents.

Table 3: Six Configurations of MLRs

	anaphoric chain	anaphoric type identification	confidence factor
MLR-1	yes	no	100%
MLR-2	yes	no	75%
MLR-3	yes	no	50%
MLR-4	yes	no	25%
MLR-5	yes	yes	75%
MLR-6	no	no	75%

unlikely hypotheses from the set. An orderer ranks hypotheses in a preference order if there is more than one hypothesis left in the set after applying all the applicable filters. Table 4 shows KS's employed for the four anaphoric types.

3.3 Scoring Method

We used *recall* and *precision* metrics, as shown in Table 5, to evaluate the performance of anaphora resolution. It is important to use both measures because one can build a high recall–low precision system or a low recall–high precision system, neither of which may be appropriate in certain situations. The NLP system sometimes fails to create discourse markers exactly corresponding to anaphora in texts due to failures of lexical or syntactic processing. In order to evaluate the performance of the anaphora resolution systems themselves, we only considered anaphora whose discourse markers were identified by the NLP system in our evaluation. Thus, the system performance evaluated against *all* the anaphora in texts could be different.

Table 5: Recall and Precision Metrics for Evaluation

Recall = N_c/I, Precision = N_c/N_h	
I	Number of system-identified anaphora in input
N_c	Number of correct resolutions
N_h	Number of resolutions attempted

3.4 Testing Results

The testing was done using 1359 anaphora (of which 1271 were one of the four anaphoric types) in 200 blind test texts for both the MLRs and the MDR. It should be noted that both the training and testing texts are newspaper articles about joint ventures, and that each article always talks about more than one organization. Thus, finding antecedents of organizational anaphora is not straightforward. Table 6 shows the results of six different MLRs and the MDR for the four types of anaphora, while Table 7 shows the results of the MLR-2 with different sizes of training examples.

4 Evaluation

4.1 The MLRs vs. the MDR

Using F-measures[5] as an indicator for overall performance, the MLRs with the chain parameters turned on and type identification turned off (i.e. MLR-1, 2, 3, and 4) performed the best. MLR-1, 2, 3, 4, and 5 all exceeded the MDR in overall performance based on F-measure.

Both the MLRs and the MDR used the character subsequence, the proper noun category, and the semantic class feature values for NAME-ORG anaphora (in MLR-5, using anaphoric type identification). It is interesting to see that the MLR additionally uses the topicalization feature before testing the semantic class feature. This indicates that, information theoretically, if the topicalization feature is present, the semantic class feature is not needed for the classification. The performance of NAME-ORG is better than other anaphoric phenomena because the character subsequence feature has very high antecedent predictive power.

4.1.1 Evaluation of the MLRs

Changing the three parameters in the MLRs caused changes in anaphora resolution performance. As Table 6 shows, using anaphoric chains without anaphoric type identification helped improve the MLRs. Our experiments with the confidence factor parameter indicates the trade off between recall and precision. With 100% confidence factor, which means no pruning of the tree, the tree overfits the examples, and leads to spurious uses of features such as the number of sentences between an anaphor and an antecedent near the leaves of the generated tree. This causes the system to attempt more anaphor resolutions albeit with lower precision. Conversely, too much pruning can also yield poorer results.

MLR-5 illustrates that when anaphoric type identification is turned on the MLR's performance drops

[5] F-measure is calculated by:

$$F = \frac{(\beta^2 + 1.0) \times P \times R}{\beta^2 \times P + R}$$

where P is precision, R is recall, and β is the relative importance given to recall over precision. In this case, β = 1.0.

Table 4: KS's used by the MDR

	Generators	Filters	Orderers
NAME-ORG	current-text	syntactic-category-propn name-char-subsequence semantic-class-org	reverse-recency
DNP-ORG	current-text	semantic-class-org semantic-amount-singular	topicalization subject-np recency
QZPRO-ORG	current-paragraph	not-in-the-same-dc semantic-class-from-pred	topicalization subject-np category-np recency
ZPRO-ORG	current-paragraph	not-in-the-same-dc semantic-class-from-pred	topicalization subject-np category-np recency

Table 6: Recall and Precision of the MLRs and the MDR

	NAME-ORG		DNP-ORG		QZPRO-ORG		ZPRO-ORG		Average		F-measure
# exmpls	631		54		383		203		1271		1271
	R	P	R	P	R	P	R	P	R	P	F
MLR-1	84.79	92.24	44.44	50.00	65.62	80.25	40.78	64.62	70.20	83.49	76.27
MLR-2	84.79	93.04	44.44	52.17	64.84	84.69	39.32	73.64	69.73	86.73	77.30
MLR-3	83.20	94.09	37.04	58.82	63.02	84.91	35.92	73.27	67.53	88.04	76.43
MLR-4	83.84	94.30	38.89	60.00	64.06	85.12	37.86	76.47	68.55	88.55	77.28
MLR-5	85.74	92.80	44.44	55.81	56.51	89.67	15.53	78.05	63.84	89.55	74.54
MLR-6	68.30	91.70	29.63	64.00	54.17	90.83	13.11	75.00	53.49	89.74	67.03
MDR	76.39	90.09	35.19	50.00	67.19	67.19	43.20	43.20	66.51	72.91	69.57

Table 7: MLR-2 Configuration with Varied Training Data Sizes

	NAME-ORG		DNP-ORG		QZPRO-ORG		ZPRO-ORG		Average		F-measure
# texts	R	P	R	P	R	P	R	P	R	P	F
50	81.30	91.94	35.19	48.72	59.38	76.77	29.13	56.07	64.31	81.92	72.06
100	82.09	92.01	38.89	53.85	63.02	85.82	28.64	62.77	65.88	85.89	74.57
150	82.57	91.89	48.15	60.47	55.73	85.60	20.39	70.00	62.98	87.28	73.17
200	83.99	91.70	46.30	60.98	63.02	82.88	36.41	65.22	68.39	84.99	75.79
250	84.79	93.21	44.44	53.33	65.10	83.89	40.78	73.04	70.04	86.53	77.42
295	84.79	93.04	44.44	52.17	64.84	84.69	39.32	73.64	69.73	86.73	77.30
MDR	76.39	90.09	35.19	50.00	67.19	67.19	43.20	43.20	66.51	72.91	69.57

but still exceeds that of the MDR. MLR-6 shows the effect of not training on anaphoric chains. It results in poorer performance than the MLR-1, 2, 3, 4, and 5 configurations and the MDR.

One of the advantages of the MLRs is that due to the number of different anaphoric types present in the training data, they also learned classifiers for several additional anaphoric types beyond what the MDR could handle. While additional coding would have been required for each of these types in the MDR, the MLRs picked them up without additional work. The additional anaphoric types included DPRO, REFLEXIVE, and TIMEI (cf. Table 1). Another advantage is that, unlike the MDR, whose features are hand picked, the MLRs automatically select and use necessary features.

We suspect that the poorer performance of ZPRO-ORG and DNP-ORG may be due to the following deficiency of the current MLR algorithms: Because anaphora resolution is performed in a "batch mode" for the MLRs, there is currently no way to percolate the information on an anaphor-antecedent link found by a system after each resolution. For example, if a zero pronoun (Z-2) refers to another zero pronoun (Z-1), which in turn refers to an overt NP, knowing which is the antecedent of Z-1 may be important for Z-2 to resolve its antecedent correctly. However, such information is not available to the MLRs when resolving Z-2.

4.1.2 Evaluation of the MDR

One advantage of the MDR is that a tagged training corpus is not required for hand-coding the resolution algorithms. Of course, such a tagged corpus is necessary to evaluate system performance quantitatively and is also useful to consult with during algorithm construction.

However, the MLR results seem to indicate the limitation of the MDR in the way it uses orderer KS's. Currently, the MDR uses an ordered list of multiple orderer KS's for each anaphoric type (cf. Table 4), where the first *applicable* orderer KS in the list is used to pick the best antecedent when there is more than one possibility. Such selection ignores the fact that even anaphora of the same type may use different orderers (i.e. have different preferences), depending on the types of possible antecedents and on the context in which the particular anaphor was used in the text.

4.2 Training Data Size vs. Performance

Table 7 indicates that with even 50 training texts, the MLR achieves better performance than the MDR. Performance seems to reach a plateau at about 250 training examples with a F-measure of around 77.4.

5 Related Work

Anaphora resolution systems for English texts based on various machine learning algorithms, including a decision tree algorithm, are reported in Connolly *et al.* (Connolly et al., 1994). Our approach is different from theirs in that their decision tree identifies which of the two possible antecedents for a given anaphor is "better". The assumption seems to be that the closest antecedent is the "correct" antecedent. However, they note a problem with their decision tree in that it is not guaranteed to return consistent classifications given that the "preference" relationship between two possible antecedents is not transitive.

Soderland and Lehnert's machine learning-based information extraction system (Soderland and Lehnert, 1994) is used specifically for filling particular templates from text input. Although a part of its task is to merge multiple referents when they corefer (i.e. anaphora resolution), it is hard to evaluate how their anaphora resolution capability compares with ours, since it is not a separate module. The only evaluation result provided is their extraction result. Our anaphora resolution system is modular, and can be used for other NLP-based applications such as machine translation. Soderland and Lehnert's approach relies on a large set of filled templates used for training. Domain-specific features from those templates are employed for the learning. Consequently, the learned classifiers are very domain-specific, and thus the approach relies on the availability of new filled template sets for porting to other domains. While some such template sets exist, such as those assembled for the Message Understanding Conferences, collecting such large amounts of training data for each new domain may be impractical.

Zero pronoun resolution for machine translation reported by Nakaiwa and Ikehara (Nakaiwa and Ikehara, 1992) used only semantic attributes of verbs in a restricted domain. The small test results (102 sentences from 29 articles) had high success rate of 93%. However, the input was only the first paragraphs of newspaper articles which contained relatively short sentences. Our anaphora resolution systems reported here have the advantages of domain-independence and full-text handling without the need for creating an extensive domain knowledge base.

Various theories of Japanese zero pronouns have been proposed by computational linguists, for example, Kameyama (Kameyama, 1988) and Walker *et al.* (Walker et al., 1994). Although these theories are based on dialogue examples rather than texts, "features" used by these theories and those by the decision trees overlap interestingly. For example, Walker *et al.* proposes the following ranking scheme to select antecedents of zero pronouns.

(GRAMMATICAL or ZERO) TOPIC > EMPATHY >
SUBJECT > OBJECT2 > OBJECT > OTHERS

In examining decision trees produced with anaphoric type identification turned on, the following features were used for QZPRO-ORG in this order: topicalization, distance between an anaphor and an antecedent, semantic class of an anaphor and an antecedent, and subject NP. We plan to analyze further the features which the decision tree has used for zero pronouns and compare them with these theories.

6 Summary and Future Work

This paper compared our automated and manual acquisition of anaphora resolution strategies, and reported optimistic results for the former. We plan to continue to improve machine learning-based system performance by introducing other relevant features. For example, discourse structure information (Passonneau and Litman, 1993; Hearst, 1994), if obtained reliably and automatically, will be another useful domain-independent feature. In addition, we will explore the possibility of combining machine learning results with manual encoding of discourse knowledge. This can be accomplished by allowing the user to interact with the produced classifiers, tracing decisions back to particular examples and allowing users to edit features and to evaluate the efficacy of changes.

References

Chinatsu Aone and Scott W. Bennett. 1994. Discourse Tagging Tool and Discourse-tagged Multilingual Corpora. In *Proceedings of International Workshop on Sharable Natural Language Resources (SNLR)*.

Chinatsu Aone and Douglas McKee. 1993. Language-Independent Anaphora Resolution System for Understanding Multilingual Texts. In *Proceedings of 31st Annual Meeting of the ACL*.

Chinatsu Aone, Sharon Flank, Paul Krause, and Doug McKee. 1993. SRA: Description of the SOLOMON System as Used for MUC-5. In *Proceedings of Fourth Message Understanding Conference (MUC-5)*.

Chinatsu Aone. 1994. Customizing and Evaluating a Multilingual Discourse Module. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING)*.

Susan Brennan, Marilyn Friedman, and Carl Pollard. 1987. A Centering Approach to Pronouns. In *Proceedings of 25th Annual Meeting of the ACL*.

Dennis Connolly, John D. Burger, and David S. Day. 1994. A Machine Learning Approach to Anaphoric Reference. In *Proceedings of International Conference on New Methods in Language Processing (NEMLAP)*.

Marti A. Hearst. 1994. Multi-Paragraph Segmentation of Expository Text. In *Proceedings of 32nd Annual Meeting of the ACL*.

Jerry R. Hobbs. 1976. Pronoun Resolution. Technical Report 76-1, Department of Computer Science, City College, City University of New York.

Megumi Kameyama. 1988. Japanese Zero Pronominal Binding, where Syntax and Discourse Meet. In *Papers from the Second International Worksho on Japanese Syntax*.

Hans Kamp. 1981. A Theory of Truth and Semantic Representation. In J. Groenendijk et al., editors, *Formal Methods in the Study of Language*. Mathematical Centre, Amsterdam.

Hiromi Nakaiwa and Satoru Ikehara. 1992. Zero Pronoun Resolution in a Japanese to English Machine Translation Systemby using Verbal Semantic Attribute. In *Proceedings of the Fourth Conference on Applied Natural Language Processing*.

Rebecca J. Passonneau and Diane J. Litman. 1993. Intention-Based Segmentation: Human Reliability and Correlation with Linguistic Cues. In *Proceedings of 31st Annual Meeting of the ACL*.

J. Ross Quinlan. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers.

Stephen Soderland and Wendy Lehnert. 1994. Corpus-driven Knowledge Acquisition for Discourse Analysis. In *Proceedings of AAAI*.

Marilyn Walker, Masayo Iida, and Sharon Cote. 1994. Japanese Discourse and the Process of Centering. *Computational Linguistics*, 20(2).

Marilyn A. Walker. 1989. Evaluating Discourse Processing Algorithms. In *Proceedings of 27th Annual Meeting of the ACL*.