A Machine Learning Approach to Identification and Resolution of *One*-Anaphora

Hwee Tou Ng*, Yu Zhou*, Robert Dale† and Mary Gardiner†
*Department of Computer Science, National University of Singapore, Singapore
†Centre for Language Technology, Macquarie University, Sydney, Australia
{nght,zhouyu}@comp.nus.edu.sg, {rdale,gardiner}@ics.mq.edu.au

Abstract

We present a machine learning approach to identifying and resolving *one*-anaphora. In this approach, the system first learns to distinguish different uses of instances of the word *one*; in the second stage, the antecedents of those instances of *one* that are classified as anaphoric are then determined. We evaluated our approach on written texts drawn from the informative domains of the British National Corpus (BNC), and achieved encouraging results. To our knowledge, this is the first learning-based system for the identification and resolution of *one*-anaphora.

1 Introduction

The word *one* is a frequently used word in English: in the 100-million-word British National Corpus (BNC), it accounts for 0.26% of all the words, ranking as the 110th most common word, and the 14th most common pronominal form, more frequent than the pronoun *us*.[1] Not all of these instances of *one* are anaphoric; the most common use is as a number (as in *one book*), and there are a number of other uses. However, any natural language processing (NLP) system which tries to process anaphora in unrestricted text will need to be able to determine whether a particular instance of *one* is being used anaphorically, and when this is the case, what the antecedent of the anaphor is.

In brief, a *one*-anaphor is an anaphoric noun phrase (NP) headed by the word *one*, as in the following example:

(1) Her greater sympathy for *the Atlantic connection* over *the European one* is not widely shared by colleagues.

One-anaphora is sometimes referred to as “identity-of-sense anaphora”, in contrast to the more common pronominal “identity-of-reference” anaphora [Hankamer and Sag, 1976]. In processing a pronominal reference, we are generally looking for an antecedent noun phrase that refers to the same entity as the anaphoric form; however, *one*-anaphoric forms are generally used to refer to something of the same *kind* as something mentioned before. Hence, in order to interpret an instance of *one*-anaphora, we must identify the antecedent textual material that provides the semantic content alluded to by the use of *one*. In Example (1), correctly interpreting *the European one* requires inferring that it has a relationship to the noun phrase *the Atlantic connection*. The range of semantic relationships that can hold between an *one*-anaphoric form and its antecedent is broad and complex; for present purposes, we are interested in identifying the antecedent noun phrase that contains the relevant semantic content.

We designed two systems that are applied consecutively to identify and resolve *one*-anaphora. In the case of Example (1), our *one*-expression classifier first determines that *the European one* is in fact a *one*-anaphor; then our *one*-anaphora resolver would identify the noun phrase *the Atlantic connection* as the antecedent of the *one*-anaphor. In this paper, an antecedent is defined as the noun phrase from which the head sense of the *one*-anaphor can be determined.

The pervasiveness of anaphoric reference in general means that anaphora resolution is recognized as an important subtask in natural language processing; *one*-anaphora resolution, however, has been relatively neglected. As noted, the frequency of occurrence of the word *one* means that any real NLP system cannot just ignore it; however, the knowledge sources (typically, features of the linguistic context) used in state-of-the-art noun phrase coreference resolution systems (e.g., [Soon *et al.*, 2001; Ng and Cardie, 2002]) are not applicable to *one*-anaphora. Consequently, *one*-anaphora resolution is a task that requires special attention.

The rest of this paper is organized as follows. We first introduce the various uses of the word *one* in English and the different types of antecedents of *one*-anaphora. Then we give an overview of how our *one*-expression classifier and *one*-anaphora resolver are applied consecutively to identify and resolve *one*-anaphora, followed by two separate sections giving a detailed description and evaluation of the two systems. In the final two sections, we describe related work and conclude.

2 Classes of *One*-Expressions

In this section, we introduce the various uses of *one* in English and the different types of antecedents of *one*-anaphora, along with some statistics of their distribution in the BNC.

[1]These figures are based on Adam Kilgarriff's frequency lists at http://www.itri.brighton.ac.uk/∼Adam.Kilgarriff/bnc-readme.html.

2.1 Uses of *One* and Corpus Annotation

The taxonomy of uses of the word *one* adopted in this section is based on existing theories of the various uses of *one* [Halliday and Hasan, 1976; Webber, 1979; Dahl, 1985; Luperfoy, 1991] and our own study of *one*-expressions. We divided uses of *one* into six classes: Numeric, Partitive, Anaphoric, Generic, Idiomatic, and Unclassifiable.

1. **Numeric** ***One*****:** Modifies a head noun to indicate singularity as in Example (2); this is the only adjectival use of *one*.

(2) John has *one* blue T-shirt.

2. **Partitive** ***One*****:** Selects an individual from a set. It is followed by an *of*-prepositional phrase headed by a plural noun or pronoun as in Example (3).

(3) A special exhibition of books for children forms *one* of the centrepieces of the 41st annual Frankfurt Book Fair.

3. **Anaphoric** ***One*****:** Relates a set of properties to the set of properties mentioned by the antecedent. There are three types of *one*-anaphors, distinguished by the type of their antecedents. First, the antecedent may be a kind, as in the *noisy cameras* of Example (4); second, the antecedent may be a set of entities, as in the set of *two World Bank men* of Example (5); and third, the antecedent may refer to a single instance, as in *this book* of Example (6).

(4) I have an aversion to *noisy cameras*, and this *one* rings several decibels before it's done with winding on the film.

(5) *The two World Bank men*, *one* German and *one* British, strode across the tarmac.

(6) Would you like *this book*? Yes, I would like that *one*.

4. **Generic** ***One*****:** A pronominal use that refers to a generic person or to the speaker of a sentence; often used in subject position followed by a modal verb and a main verb that takes an animate subject.

(7) *One* must think a little deeper to discover the underlying social roots of the problem.

5. **Idiomatic** ***One*****:** Conventionalized uses whose semantics appear not to be based on general use, but rather on idiomatic patterns, as in Example (8).

(8) It would be perfect to have a loved *one* accompany me in the whole trip.

6. **Unclassifiable** ***One*****:** Inevitably, there are instances which are difficult to classify as any of the above, as in Example (9).

(9) Cursed be every one who curses you.

For the present study, we randomly selected 1,577 *one* expressions from the BNC [2], and manually annotated these with the six classes above. The distribution of each class in the annotated corpus is shown in Table 1, and it mirrors the distribution of *one* expressions in naturally occurring text.

Class	Frequency	%
Numeric	739	46.9
Partitive	399	25.3
Anaphoric	240	15.2
Generic	167	10.6
Unclassifiable	25	1.6
Idiomatic	7	0.4
Total	1,577	100.0

Table 1: Distribution of uses of *one* in the annotated corpus.

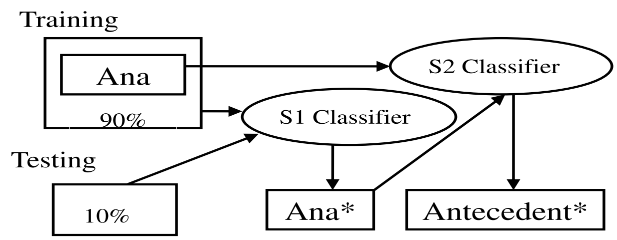

Figure 1: How training and test data are used.

2.2 Annotating Antecedents of *One*-Anaphora

We annotated the 240 examples of *one*-anaphors in our corpus a second time, marking the antecedent in each case. Not every *one*-anaphor has an explicit noun phrase antecedent as shown in the previous anaphoric examples; in such cases, the reader has to infer the nature of the antecedent from the available text. We label those cases where there is no explicit antecedent as "*one*-anaphors with implicit antecedents". The remaining *one*-anaphors are "*one*-anaphors with explicit antecedents". Of the 240 *one*-anaphors, 98.3% had explicit antecedents.

3 An Overview of Our Approach

Before we describe our two machine learning systems for *one*-expression classification and *one*-anaphora resolution, we first provide an overview of how the two systems are applied consecutively to accomplish the overall task of identifying and resolving *one*-anaphora. As an example, we use one trial in the 10-fold cross validation to illustrate the process. Here, the identification of *one*-anaphora is Step 1 (S1), and the resolution of *one*-anaphora is Step 2 (S2).

As shown in Figure 1, a gold standard corpus is divided into two sets, containing 90% and 10% of the data respectively. The larger set is used to train the S1 classifier. In the S1 training data, there is a set of anaphoric examples, Ana. This set is used to train the S2 classifier.

The smaller set of 10% of the examples is used for testing. First, the test examples are passed to the S1 classifier, which identifies a set of anaphoric examples, Ana*. This set may contain errors in identification (a *one*-anaphor not classified as anaphoric, or a non-anaphoric example classified as

[2] These *one*-expressions are from written text in the informative domains of BNC. Spoken text and written text in the imaginative domain (i.e., texts which are fictional or which are generally perceived as literary or creative) are not considered because they contain a large amount of dialog, which makes *one*-anaphora resolution a harder task. This restriction follows the genre of MUC texts [MUC-7, 1997], a widely used data set in noun phrase coreference research, which also only included written newspaper texts.

anaphoric). Ana* is in turn passed to the S2 classifier, which outputs the antecedents it finds, denoted as Antecedent*. When calculating accuracy, Ana* and Antecedent* are used to compare with the gold standard annotation to measure both the S1 accuracy and the overall accuracy of the combined systems.

4 *One*-Expression Classification

Our *one*-expression classifier classifies a given *one*-expression into one of the six classes listed in Table 1.

4.1 The Features

We devised a set of features that is useful in determining which of the six classes a given *one*-expression belongs to. We focus on the classes Numeric, Partitive, Anaphoric, and Generic, since 98% of the instances are from these classes.

The Numeric use of *one* is the simplest case, since it is the only adjectival use in English. It can be readily identified by the part-of-speech (POS) tag of the word *one*. Partitive use of *one* can be identified by checking the syntactic context (i.e., [*one* [*of* NP_{pl}]]).

Discrimination between the Anaphoric and Generic classes is more complex. We used the three features *isSubj*, *isAnimateVerb*, and *isModalVerb* to identify instances of the Generic class, in line with the observations mentioned in the definition of Generic *One* . We also noticed that the relative position of *one* in its host NP and the category of the word immediately preceding *one* both give strong hints as to the function of *one* in its host NP. Therefore, we added two more features, *positionInNP* and $W_{-1}POS$, to assist in classification.

We experimented with a total of 7 features as described below. Each instance in the training and test data set is thus represented as a feature vector of seven values. The feature values are acquired by running the Charniak Parser [Charniak, 2000] over the corpus, in combination with information about verbs of cognition from WordNet [Fellbaum, 1998]:

1. **W_0POS** is the POS tag of *one* assigned by the Charniak Parser. Its possible values are *CD, NN, NNP, PRP*.
2. **isOfPlural** checks whether *one* is followed by *of* prepositional phrase with a plural head noun/pronoun. Its possible values are *plural*, *notPlural*, or *NA* when the word *one* is not followed by a PP headed by *of*. For example:
   (a) *plural*: *One* of the patients survived.
   (b) *notPlural*: The problem was the unusual *one* of a warmish, wet spring.
3. **isSubj** checks whether the word *one* is in subject position. This feature value is inferred from the parse tree structure; its possible values are *true* or *false*.
4. **isAnimateVerb** checks whether the lemmatized verb following subject *one* is a verb of cognition according to WordNet: some examples are *think*, *judge*, *analyze*, and *doubt*. The possible values for this feature are *true*, *false*, or *NA* when *one* is not the subject. Example (7) would get *true* for this feature.
5. **isModalVerb** checks whether the verb phrase following a subject *one* contains a modal verb (i.e., a word with MD POS tag, such as *must*). The possible values for this feature are *true*, *false*, or *NA* when *one* is not the subject. Example (7) would get *true* for this feature.
6. **positionInNP** indicates the position of the word *one* in its host NP. It has four possible values: *SingleOne* when the NP only consists of the word *one*; *Leftmost* when the NP has multiple words and *one* is the leftmost word; *Rightmost* when the NP has multiple words and *one* is the rightmost word; and *Middle* otherwise. For example:
   a) *SingleOne*: [*One*] might suppose that ...
   b) *Leftmost*: I was concerned with [*one* thing] only.
   c) *Rightmost*: [the European *one*]
   d) *Middle*: on [the *one* hand]
7. **W_{-1}POS** is the POS tag of the word immediately preceding *one*. Its possible values are the 45 POS tags in Penn TreeBank tagset and *NA* when *one* is the first word of the sentence.

	Num in data set	Num identified	Num correct	R	P	F1
NUM	739	730	693	93.8	94.9	94.3
PAR	399	412	390	97.7	94.7	96.2
ANA	240	281	192	80.0	68.3	73.7
GEN	167	152	127	76.0	83.6	79.6
Sub Total	1,545	1,575	1,402	90.7	89.0	89.8
UNC	25	2	0	0	0	—
IDIO	7	0	0	0	—	—
Total	1,577	1,577	1,402	88.9	88.9	88.9

Table 2: Accuracy of *one*-expression classification.

The learning algorithm used in our *one*-expression classification system is C4.5 [Quinlan, 1993]. This is a commonly used decision tree learning algorithm and may be considered as a baseline against which other learning algorithms can be compared.

4.2 Evaluation and Error Analysis

Evaluation

Table 2 gives the 10-fold cross validation results for the *one*-expression classifier for each of the 6 classes.

The accuracy of identifying *one*-anaphora is 73.7%, which is not as high as the Numeric, Partitive, and Generic classes. The remaining two classes, Idiomatic and Unclassifiable, are poorly discriminated; when counted in, these pull down the overall classification accuracy from 89.8% to 88.9%.

Error Analysis

Table 3 provides a matrix of the number of misclassifications in each class. Row 4 and Column 4, highlighted, show that the major confusion with the Anaphoric class comes from the Generic and Numeric classes, with Partitive making a smaller contribution to the erroneous classifications.

Our classifier's performance is highly impacted by the accuracy of POS tagging and parsing: failures here caused most of the confusion with Numeric and Partitive. For example, *one* in (4) is classified as Numeric because *rings* is wrongly

classified as=>*	NUM*	PAR*	**ANA***	GEN*	UNC*	IDIO*
NUM	na	2	**35**	8	1	0
PAR	3	na	**6**	0	0	0
ANA	**14**	**18**	**na**	**15**	**1**	**0**
GEN	9	0	**31**	na	0	0
UNC	10	1	**12**	2	na	0
IDIO	1	1	**5**	0	0	na

Table 3: *One*-expression classifier error matrix.

tagged as NNS; and Partitive *one* in the phrase *one or more of them* is classified as Anaphoric because it is wrongly parsed as *[[one] or [more of [them]]]*.

The confusion with the Generic class is mainly caused by occurrences of anaphoric *one* as subject and generic *one* in non-subject position. Such cases can also be confusing for human readers: in a sentence like *One is usually shunting around in the yard*, we might think we have a generic use, but the *one* may refer to a previously mentioned *locomotive*.

5 *One*-Anaphora Resolution

Our *one*-anaphor resolver attempts to identify the NP in the preceding linguistic context that provides the semantic content required for interpretation of a *one*-anaphor; we refer to this NP as the antecedent NP.

5.1 Experimental Data

We trained and tested a *one*-anaphor resolution classifier using a set of positive/negative antecedent-anaphor pairs. A pair is positive when the candidate antecedent in this pair is the actual antecedent of the anaphor; otherwise, it is negative.

Training Data

In each trial of the 10-fold cross validation, the actual *one*-anaphora instances in the gold standard training corpus of Step 1 are used to create Step 2 training instances in the corresponding trial as shown in Figure 1.

Creation of training instances consisted of three steps. First, each sentence containing *one* and its three preceding sentences [3] were processed by RM NP chunker [Ramshaw and Marcus, 1995] to carry out NP chunking. Second, each pair of an anaphor and its actual antecedent were used to create a positive training instance. Lastly, to generate negative training instances, anaphors were paired with each of the NPs that appeared between the anaphor and its real antecedent.

In our experiment, we further adjusted the ratio of positive to negative instances in the training data by controlling the number of negative instances randomly picked from the whole set of negative instances. We decided to set the ratio at 1:1, which introduces no preference for either one of the two assignments to the classifier. This procedure produced a set of 472 antecedent-anaphor pairs in total, of which 236 (50%) were positive instances. As already noted, each trial used roughly 90% of this set as training data.

[3] If the sentence containing one appeared close to the beginning of the text and there were less than three preceding sentences, we used all available preceding sentences.

Test Data

In each trial, instances of *one* identified as being anaphoric in the test data of Step 1 were used to create Step 2 test instances in the corresponding trial as shown in Figure 1.

Creation of test instances consisted of two steps. The first step is the same as that for the training data; in the second step, every base NP preceding the instance of *one* is a potential antecedent, so each of these NPs was paired with *one*.

When doing testing, the *one*-anaphora resolution algorithm starts from the immediately preceding base NP and proceeds backward in the reverse order of the NPs in the context until there is no remaining NP to test, or an antecedent is found (i.e., the classifier returns true).

5.2 The Features

To decide whether the anaphor in a given antecedent-anaphor pair refers to the candidate antecedent, we need features that show a preference for good candidates.

As with pronominal anaphora, an intuitively appealing feature to use is whether the candidate antecedent NP is in focus [Sidner, 1981; Vieira and Poesio, 2000]. In anaphora resolution, two commonly used features for approximating the notion of focus are syntactic role and recency: a candidate that fills a salient syntactic role such as subject, or one that is very recent, is often the focus of the discourse. We used four features to measure this type of information: *AnteIsSubj*, *AnteInRelClause*, *AnteIsNearestNP*, and *bothInPP*. The feature *bothInPP* allows us to take syntactic parallelism into consideration. The POS tag of the head word of the candidate antecedent is also a good feature in filtering out improper candidates: a proper noun should not be the antecedent of a *one*-anaphor [Dahl, 1985].

We used a total of 5 features as described below. Their values are acquired from the Charniak Parser output of the training/test data set.

1. **hwPOSofAnte** is the POS tag of the head word of the candidate antecedent. Its possible values are the set of POS tags of head words that occur in our training/test data. In this paper, the head word of the candidate antecedent is defined as the rightmost noun, or the rightmost word if no noun is found, in the base NP candidate antecedent.

2. **bothInPP** checks whether the candidate antecedent and the *one*-anaphor are both in prepositional phrases (PP), and identifies the types of PPs they are in. It has five possible values: *NA* when the candidate antecedent is not in PP; *OnlyAnteInPP* when the candidate antecedent is in PP, but the *one*-anaphor is not in PP; *SharePP* when the candidate antecedent and the *one*-anaphor are in the same PP; *CommonPreposition* when both candidate antecedent and *one*-anaphor are in different PPs, but the prepositions of the two PPs are the same; and *DifferentPreposition* when both candidate antecedent and *one*-anaphor are in different PPs with different prepositions.

3. **AnteIsSubj** checks whether the candidate antecedent is in subject position. Its possible values are *true* or *false*.

4. **AnteIsNearestNP** checks whether the candidate antecedent is the nearest NP preceding the *one*-anaphor. Its possible values are *true* or *false*.
5. **AnteInRelClause** checks whether the candidate antecedent is in a relative clause. Its possible values are *true* or *false*.

Again, the learning algorithm used in our one-anaphora resolution engine is C4.5.

5.3 Evaluation and Error Analysis

Evaluation

In order to evaluate the overall performance of our approach to *one*-anaphora identification and resolution, we conducted a 10-fold cross validation of the *one*-anaphora resolver, where each trial is performed based on the result of our *one*-expression classifier in Step 1. The overall recall, precision, and F-measure are presented in Table 4, together with the accuracy of *one*-anaphora identification from Step 1.

The "S2 correct" in Table 4 Column 9 is the sum of two values. The first value is the number of hits in correctly identifying the explicit antecedent of a *one*-anaphor. A hit of this type occurs when a *one*-anaphor with an explicit antecedent is correctly identified as anaphoric in Step 1 and the actual antecedent is found in Step 2. The second value is the number of hits returning no antecedent for a *one*-anaphor when it has no annotated antecedent. A hit of this type occurs when a *one*-anaphor without an explicit antecedent is correctly identified as anaphoric in Step 1, and none of the candidate antecedents is accepted in Step 2.

The overall recall is the sum in "S2 correct" divided by the total number of *one*-anaphors with or without explicit antecedent in the data set (Column 1); the overall precision is the sum in "S2 correct" divided by the total number of *one*-anaphors with or without explicit antecedent identified in Step 1 (Column 3). The F-measure finally achieved is 45.7%.

We compared two baseline heuristics with our accuracy: nearestNP and nearestSubj, which always assign the nearest NP or Subject preceding the *one*-anaphor as its antecedent. The two baseline accuracies are calculated by applying the two heuristics on the set of anaphoric *ones* identified in Step 1; nearestNP (nearestSubj) heuristics achieved 28.8% (23.8%) accuracy, considerably lower than the overall accuracy of our system.

Error Analysis and Future Improvement

Since we perform *one*-anaphora resolution on the output of our *one*-anaphora identification system, the 45.7% overall accuracy is a combination of both Step 1 and Step 2 performance. Errors introduced in Step 1 were never remedied in Step 2, and they directly affected both overall recall and precision. In other words, given a perfect Step 2 classifier to work on the current *one*-anaphora identification output, the highest overall F-measure achievable is 73.7%. Therefore, any further improvement in Step 1 performance would significantly improve the overall accuracy. This could be done by adding features to improve the performance of Step 1, or adding a remedy strategy in later processing.

Our system finds it difficult to locate actual antecedents that are far away from their *one*-anaphors: preference is wrongly given to closer and more salient candidates. In Example (10), *the director* has a strong syntactic preference and it is considered before *moustache*, so the system wrongly returns *the director* even before checking the actual antecedent.

(10) He has this ridiculous *moustache*. Ken Russell, the director, insisted I grew one of my own, rather than wear a false *one*, so that I looked completely convincing.

We expect that such mistakes could be corrected by using semantic features.

5.4 The Contribution of the Features

To evaluate the relative contribution of the various knowledge sources to the overall accuracy of *one*-anaphora identification and resolution, we ran a series of leave-one-out classifiers, where we first used all Step 1 features and disabled one Step 2 feature at a time; then we used all Step 2 features and disabled one Step 1 feature at a time. The contribution of the features measured in terms of the overall F-measure is shown in Table 4 Column 12.

The critical features which cause a substantial reduction of overall F-measure when disabled are Step 1 features *positionInNP*, *isOfPlural*, and W_0POS, as well as Step 2 features *hwPOSofAnte* and *bothInPP*. The remaining features only have a small impact on the overall F-measure.

6 Related Work

The literature on *one*-anaphora is small; we cited the most significant works in the area in Section 2. Most of the existing literature is more concerned with describing the phenomenon than in determining how it might be handled automatically.

There is, of course, an extensive literature on computational techniques for resolving pronominal anaphora, going back to at least the 1970s. Of the more recent research in the area, important work is that of Lappin and Leass [1994] and Kennedy and Boguraev [1996], who provided heuristics that could be used to determine the antecedents of pronominal forms. Soon *et al*. [2001] and Ng and Cardie [2002] used a machine learning approach for coreference resolution. However, most of the linguistic features used in the work on pronominal anaphora are not applicable to the *one*-anaphora problem. Markert *et al*. [2003] focused on the related phenomenon of *other*-anaphora.

7 Conclusion

In this paper, we have presented a machine learning approach to the identification and resolution of *one*-anaphora and achieved encouraging results; to our knowledge, this is the first learning-based system for resolving *one*-anaphora. There is scope for refinement to improve both the accuracy of identifying anaphoric uses of *one*, and of identifying the antecedent noun phrase that contains the semantic content required for interpreting the *one*-anaphor. Beyond this goal, there are more complex challenges awaiting in terms of *one*-anaphora interpretation.

Num of Ana in data set	**Step1: 7 features (f1-f7)**						**Step2: 5 features (f1-f5)**		**Overall**			**Baseline**	
	S1 feature	S1 sysout as Ana	S1 correct	S1 R	S1 P	S1 F1	S2 feature	S2 correct	R	P	F1	n-NP	n-Subj
240	ALL	281	192	80.0	68.3	73.7	ALL	119+0	49.6	42.3	**45.7**	28.8	23.8
							no f1	78+0	32.5	27.8	30.0		
							no f2	99+1	41.7	35.6	38.4		
							no f3	117+0	48.8	41.6	44.9		
							no f4	118+0	49.2	42.0	45.3		
							no f5	119+0	49.6	42.3	45.7		
	no f1	128	76	31.7	59.4	41.3	ALL	49+0	20.4	38.3	26.6	8.3	7.1
	no f2	126	91	37.9	72.2	49.7	ALL	60+0	25.0	47.6	32.8	12.5	13.8
	no f3	251	165	68.8	65.7	67.2	ALL	100+0	41.7	39.8	40.7	25.0	22.1
	no f4	282	185	77.1	65.6	70.9	ALL	114+0	47.5	40.4	43.7	27.1	23.3
	no f5	295	192	80.0	65.1	71.8	ALL	118+0	49.2	40.0	44.1	29.2	24.6
	no f6	276	187	77.9	67.8	72.5	ALL	115+0	47.9	41.7	44.6	27.5	23.3
	no f7	280	191	79.6	68.2	73.5	ALL	118+0	49.2	42.1	45.4	28.3	23.8

Table 4: Overall accuracy and baseline of identification and resolution of *one*-anaphora & contribution of the features. Step1 7 features are 1.positionInNP, 2.isOfPlural, 3.W_0 POS, 4.isAnimate, 5.W_{-1} POS, 6.isSubj, 7.isModalVerb. Step2 5 features are 1.hwPOSofAnte, 2.bothInPP, 3.AnteInRelClause, 4.AnteIsNearestNP, 5.AnteIsSubj

References

[Charniak, 2000] E. Charniak. A maximum-entropy inspired parser. In *Proceedings of NAACL*, pages 132–139, 2000.

[Dahl, 1985] D. A. Dahl. *The structure and function of one-anaphora in English*. PhD thesis, Univ. of Minnesota, 1985.

[Fellbaum, 1998] C. Fellbaum, editor. *WordNet: An electronic lexical database*. The MIT Press, 1998.

[Halliday and Hasan, 1976] M. A. K. Halliday and R. Hasan. *Cohesion in English*. Longman, 1976.

[Hankamer and Sag, 1976] J. Hankamer and I. Sag. Deep and surface anaphora. *Linguistic Inquiry*, 7:391–428, 1976.

[Kennedy and Boguraev, 1996] C. Kennedy and B. Boguraev. Anaphora for everyone: pronominal anaphora resolution without a parser. In *Proceedings of COLING*, pages 113–118, Copenhagen, Denmark, 1996.

[Lappin and Leass, 1994] S. Lappin and H. Leass. A syntactically based algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20:535–561, 1994.

[Luperfoy, 1991] S. Luperfoy. *Discourse Pegs: A Computational Analysis of Context-Dependent Referring Expressions*. PhD thesis, Univ. of Texas at Austin, 1991.

[Markert *et al.*, 2003] K. Markert, M. Nissim, and N. Modjeska. Using the web for nominal anaphora resolution. In *Proceedings of the EACL Workshop on Computational Treatment of Anaphora*, pages 39–46, 2003.

[MUC-7, 1997] MUC-7. Coreference task definition (v3.0, 13 Jul 97). In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, 1997.

[Ng and Cardie, 2002] V. Ng and C. Cardie. Improving machine learning approaches to coreference resolution. In *Proceedings of ACL*, 2002.

[Quinlan, 1993] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Francisco, 1993.

[Ramshaw and Marcus, 1995] L. Ramshaw and M. Marcus. Text chunking using transformation-based learning. In *Proceedings of the ACL Third Workshop on Very Large Corpora*, pages 82–94, 1995.

[Sidner, 1981] C. L. Sidner. Focusing for interpretation of pronouns. *American Journal of Computational Linguistics*, 7(4):217–231, 1981.

[Soon *et al.*, 2001] W. M. Soon, H. T. Ng, and D. Lim. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521–544, 2001.

[Vieira and Poesio, 2000] R. Vieira and M. Poesio. An empirically based system for processing definite descriptions. *Computational Linguistics*, 26(4):539–593, 2000.

[Webber, 1979] B. Webber. *A Formal Approach to Discourse Anaphora*. Garland Publishing Inc., 1979.