A Discriminative Approach to Japanese Zero Anaphora Resolution with Large-scale Lexicalized Case Frames

Ryohei Sasano
Precision and Intelligence Laboratory
Tokyo Institute of Technology
sasano@pi.titech.ac.jp

Sadao Kurohashi
Graduate School of Informatics
Kyoto University
kuro@i.kyoto-u.ac.jp

Abstract

We present a discriminative model for Japanese zero anaphora resolution that simultaneously determines an appropriate case frame for a given predicate and its predicate-argument structure. Our model is based on a log linear framework, and exploits lexical features obtained from a large raw corpus, as well as non-lexical features obtained from a relatively small annotated corpus. We report the results of zero anaphora resolution on Web text and demonstrate the effectiveness of our approach. In addition, we also investigate the relative importance of each feature for resolving zero anaphora in Web text.

1 Introduction

Zero anaphora resolution is the task of detecting and identifying the omitted arguments of a predicate. Since arguments are often omitted in Japanese, zero anaphora resolution plays an important role in the analysis of Japanese predicate-argument structures. For example, in the following text:

(i) Musuko-wa itazura-ga sukide son-TOP mischief-NOM like watashi-mo (φ-ni) te-wo yaiteiru. I φ-DAT hands-ACC burn (have difficulty)
(My son likes mischief, so I have difficulty with φ.)

the dative argument of the predicate ‘yaku’ (burn) has been omitted. The omitted element is called a zero pronoun, and in this example it refers to ‘musuko (son).’ Although most previous work has focused on zero anaphora in newspaper articles, this paper aims to resolve zero anaphora in Web text, since this involves a wider range of writing styles and is thus considered to be a more practical setting.

Reference resolution systems generally require a variety of information sources ranging from syntactic and discourse preferences to semantic preferences (Ng and Cardie, 2002; Haghighi and Klein, 2010). Since syntactic and discourse preferences are not word-specific, they can be learned from a relatively small annotated corpus. Semantic preferences, on the other hand, represent highly lexicalized knowledge, and hence it is difficult to learn these from a small annotated corpus. In some cases, knowledge of the relations between a predicate and its particular argument is insufficient, particularly for zero anaphora resolution. For example, although the dative argument of the predicate ‘yaku (bake/burn)’ is generally filled by a disk, such as a CD or DVD, it is often filled by a person, such as ‘musuko (son).’ if the accusative argument is filled by ‘te (hands)’ as in the example (i). Thus, knowledge of relations among a predicate and its multiple arguments is required to take such preferences into account.

Sasano et al. (2008) exploited large-scale case frames that were automatically constructed from 1.6 billion Web sentences as such a lexical resource, and proposed a probabilistic model for Japanese zero anaphora resolution. Their model demonstrated moderate performance, but it could not easily introduce new features, especially overlapping ones, nor take into consideration the importance of each feature, due to the assumption of independence in estimating probability. However, we think a variety of clues can be useful for zero anaphora resolution, where it is important to exploit overlapping features and consider the importance of each feature.

\(^{1}\) ‘Yaku’ is the original form of ‘yaiteiru.’

\(^{2}\) ‘Te-wo yaku’ (literally ‘burn hands’) is a Japanese idiom, which means ‘have difficulty’ in English.
Therefore, in this paper, we extend Sasano et al. (2008)’s model by incorporating it into a log-linear framework, and introduce overlapping features such as lexical features with different granularities. In addition, we also investigate the relative importance of each feature for resolving zero anaphora in Web text.

2 Related Work

Several approaches to Japanese zero anaphora resolution have been proposed. Seki et al. (2002) proposed a probabilistic model for zero pronoun detection and resolution that used hand-crafted case frames. Kawahara and Kurohashi (2004) introduced wide-coverage case frames that were automatically constructed from a large corpus to alleviate the sparseness of hand-crafted case frames. They used the case frames as selectional restrictions for zero pronoun resolution. Iida et al. (2006) explored a machine learning method using rich syntactic pattern features that represented the syntactic relations between a zero-pronoun and its candidate antecedent.

Since predicate-argument structure analysis and zero anaphora resolution are closely related, several approaches have simultaneously solved these two tasks. Sasano et al. (2008) proposed a lexicalized probabilistic model for zero anaphora resolution, which adopted an entity-mention model and simultaneously resolved predicate-argument structures and zero anaphora. Taira et al. (2008) proposed a model for analyzing predicate-argument structures by using decision lists, which integrated the tasks of semantic role labeling and zero-pronoun identification. Imamura et al. (2009) proposed a discriminative model for analyzing predicate-argument structures that simultaneously conducted zero anaphora resolution.


3 Case Frames

3.1 Lexicalized case frames

Our model exploits lexicalized case frames that are automatically constructed from 1.6 billion Web sentences by using Kawahara and Kurohashi (2002)’s method. Case frames are constructed for each predicate like PropBank frames (Palmer et al., 2005), and for each meaning of the predicate like FrameNet frames (Fillmore et al., 2003). However, neither pseudo-semantic role labels such as Arg1 in PropBank nor information about frames defined in FrameNet are included in the case frames. Each case frame describes surface cases that each predicate has and examples that can fill a case slot, which is fully-lexicalized like the subcategorization lexicon VALEX (Korhonen et al., 2006).

Note that case frames offer not only knowledge of the relations between a predicate and its particular case slot, but also knowledge of the relations among a predicate and its multiple case slots. Table 1 shows examples of constructed case frames. A different case frame is constructed for each meaning of ‘yaku (bake/burn),’ such as ‘bake foods,’ ‘have difficulty,’ and ‘burn on a disk.’

3.2 Generalization of examples

The data sparseness problem is alleviated to some extent but not eliminated by using case frames that are automatically constructed from a large corpus. For instance, there are thousands of named entities (NEs) that cannot intrinsically be covered. Sasano et al. (2008) generalized examples of case slots based on 22 common noun categories defined in the Japanese morphological analyzer JUMAN, and 8 NE classes defined by the IREX Committee (1999) to deal with this problem.

In addition, we generalized case slot examples based on automatically acquired multi-word noun clusters. Kazama and Torisawa (2008) proposed the parallelization of EM-based clustering with the aim of enabling large-scale clustering and using the resulting clusters in NE recognition. We used the resulting 2,000 clusters acquired from 1 million unique multi-word nouns.

Table 2 lists examples of the resulting 2,000 clusters. As well as common noun categories and NE classes, we calculated the average of the probabilities that each case slot example belonged to

---

3We use English examples for the sake of readability.
4http://nlp.kuee.kyoto-u.ac.jp/nl-resource/juman-e.html
Case slot | Examples: # | Generalized examples: %
---|---|---
yaku (1) | go | 1:39, owner: 26, daughter: 22, mother: 19, ... Asako: 2, etc. |
| | | [CT: PERSON]: 0.620, [NE: PERSON]: 0.116, [CL: 887]: 0.070, etc.
| | | [CT: BODY PART]: 0.075, etc.

Asako: 2, [CL: 883]: 0.221, etc.

yaku (2) | go | who: 7, teacher: 7, everyone: 5, family: 4, government: 3, etc. |
| | | [CT: PERSON]: 0.372, [NE: PERSON]: 0.128, [CT: ORGANIZATION]: 0.128, etc.

yaku (3) | go | 1: 1, husband: 1, 
| | | [CT: PERSON]: 1.000, etc.

Table 1: Example case frames for ‘yaku (bake / have difficulty / burn).’

Cluster | Nouns
---|---
CL: 32 | child (0.974), infant (0.738), kid (0.727), babies and infant (0.436), etc.
CL: 70 | CD (0.896), DVD (0.837), CD-ROM (0.603), cassette (0.512), etc.
CL: 291 | low heat (0.720), slow fire (0.715), moderate heat (0.681), distant fire (0.678), etc.
CL: 624 | dinner (0.926), supper (0.925), lunch (0.882), breakfast (0.868), etc.
CL: 883 | Chinese noodles (0.860), noodles (0.801), curry (0.793), cake (0.749), etc.
CL: 887 | mother (0.909), parents (0.875), mom (0.838), husband (0.775), father (0.774), etc.

Table 2: Examples of resulting 2,000 clusters (Kazama and Torisawa, 2008). Nouns that have high probabilities of belonging to target clusters are shown with probabilities.

4 A Discriminative Model for Zero Anaphora Resolution

4.1 Overview

Our model basically follows that of Sasano et al. (2008), except for the method of estimating possible combinations of case frames and predicate-argument structures. We also limited the target cases for zero anaphora resolution to ‘ga’ (nominative), ‘wo’ (accusative), and ‘ni’ (dative) cases. The outline of our model is as follows:

1. Parse an input text and recognize NEs.
2. Resolve coreference and link each mention to a discourse entity or create a new entity.
3. From the end of each sentence, analyze the predicate-argument structure for each verb or adjective using the following steps.
1. \(< cf = \text{‘yaku’}(1), s = \{\text{NOM:‘watashi’ (I)}, \text{ACC:‘te’ (hands)}\}, \text{DAT:‘musuko’ (son)}\>>
2. \(< cf = \text{‘yaku’}(1), s = \{\text{NOM:‘watashi’ (I)}, \text{ACC:‘te’ (hands)}\}, \text{DAT:NULL}\}>>
3. \(< cf = \text{‘yaku’}(1), s = \{\text{NOM:NULL}, \text{ACC:‘te’ (hands)}\}, \text{DAT:‘watashi’ (I)}\}>
4. \(< cf = \text{‘yaku’}(1), s = \{\text{NOM:‘musuko’ (son)}, \text{ACC:‘te’ (hands)}\}, \text{DAT:NULL}\}>
5. \(< cf = \text{‘yaku’}(2), s = \{\text{NOM:‘watashi’ (I)}, \text{ACC:‘te’ (hands)}\}, \text{DAT:‘musuko’ (son)}\}>
6. \(< cf = \text{‘yaku’}(2), s = \{\text{NOM:‘watashi’ (I)}, \text{ACC:‘te’ (hands)}\}, \text{DAT:NULL}\}>

Table 3: Examples of possible combinations of case frame \(cf\) and predicate-argument structure \(s\) for the predicate ‘yaku’ in the example (i) in Section 1. Bold font indicates the proper combination for this example.

(a) Select a case frame temporarily.
(b) List possible predicate-argument structures including omitted arguments.
(c) Estimate possible combinations of case frames and predicate-argument structures, and select the one with the highest estimate.

In 3-(b), we first consider only the overt arguments and prune away improbable structures to reduce the search space. We apply a log-linear framework to estimating a combination of a case frame and a predicate-argument structure to introduce overlapping features and take into consideration the relative importance of each feature.

Note that the estimation is not separately conducted for each argument, but for all arguments including overt and omitted arguments. For examples, when we analyze the predicate ‘yaku’ in the example (i) in Section 1, we consider the various combinations as listed in Table 3, and choose the combination with the highest estimate.

4.2 Log-linear model

When text \(t\) and target predicate \(p\) are given, we choose the combination of case frame \(cf\) and predicate-argument structure \(s\) with the highest conditional probability,

\[
(cf_{best}, s_{best}) = \arg \max_{cf,s} P(cf, s|p, t).
\]

We model the conditional probability, using a log-linear framework:

\[
P(cf, s|p, t; \Lambda) = \frac{1}{Z(p, t)} \exp\{\Lambda \cdot F(cf, s, p, t)\},
\]

\[
Z(p, t) = \sum_{\{cf, s\}\in C(p, t)} \exp\{\Lambda \cdot F(cf, s, p, t)\},
\]

where \(F = (f_1, \ldots, f_K)\) is a feature vector whose elements represent \(K\) feature functions, \(\Lambda = (\lambda_1, \ldots, \lambda_K)\) denotes a weight vector (parameter vector) for the feature functions, and \(C(p, t)\) yields a set of possible combinations of case frame \(cf\) and predicate-argument structure \(s\) for given predicate \(p\) and text \(t\).

4.3 Parameter estimation

We now describe how parameter vector \(\Lambda\) is estimated from a set of training data. When the training set consisting of \(N\) instances \(\{(s_1, p^{(1)}, t^{(1)}), (s_2, p^{(2)}, t^{(2)}), \ldots, (s^N, p^{(N)}, t^{(N)})\}\) is given, we choose the combination of \(CF\) and \(\Lambda\) that maximize the posterior probability:

\[
\max_{CF, \Lambda} \left\{ \sum_{n=1}^{N} \log P(cf^{(n)}, s^{(n)}|p^{(n)}, t^{(n)}; \Lambda) - \alpha ||\Lambda||^2 \right\},
\]

where \(\alpha\) is a regularization parameter for the L2 norm, and \(CF = (cf^{(1)}, cf^{(2)}, \ldots, cf^{(N)})\) is a combination of possible case frames, i.e., \(cf^{(n)}\) is a candidate case frame for the given instance \((s^{(n)}, p^{(n)}, t^{(n)})\). Since the appropriate case frames are not annotated in the training set, we choose an appropriate case frame in estimating parameters with the following algorithm:

1. Initialize parameter \(\Lambda\) to a random value in the range [0,1].
2. For each training instance, update \(cf^{(n)}\) that maximizes \(P(cf, s|p, t; \Lambda)\) with current parameter \(\Lambda\):

\[
\hat{cf}^{(n)} = \arg \max_{cf^{(n)}} P(cf^{(n)}, s^{(n)}|p^{(n)}, t^{(n)}; \Lambda)
\]

If \(CF = (cf^{(1)}, cf^{(2)}, \ldots, cf^{(N)})\) is not updated, we determine current parameter \(\Lambda\) as the resulting parameter.
3. If \(CF\) is updated, we renew parameter \(\Lambda\) that maximizes the posterior probability with \(N\) training instances \(\{(cf^{(1)}, s^{(1)}, p^{(1)}, t^{(1)}), \ldots, (cf^{(N)}, s^{(N)}, p^{(N)}, t^{(N)})\}\).
\[ \mathcal{L}_A = \sum_{n=1}^{N} \log P(\text{cf} | \text{yaku}(2), \text{case}=\text{ni}) / \log P(\text{child}) \]

and go back to step 2. To avoid overfitting, we include an L2-regularization term in the objective.\(^5\) \( \mathcal{L}_A \) is maximized by the Limited memory BFGS (L-BFGS) algorithm (Nocedal, 1980).\(^6\)

In both steps 2 and 3, the log-likelihood increases monotonically, and this algorithm thus always converges to an optimal solution but does not ensure the global maximum parameter will be assigned. That is, we can obtain convergence to different local optima. Therefore, we test several initial values and adopt the resulting parameter that maximizes posterior probability.

5 Features

5.1 Lexical features

We exploit six types of lexical features: word PMI (Pointwise Mutual Information), cluster PMI, category PMI, NE PMI, occupancy of a case slot, and overt argument assignment score. Their values are real values that are calculated by using case frames. Since our model is based on the entity-mention model that assigns zero pronouns not to a certain mention but to an entity, several values can be calculated for a certain lexical feature by taking coreferential mentions into consideration. In such cases, we choose the highest value as a corresponding lexical feature.

Word PMI Each case slot of a case frame has typical words that are often assigned to the slot. We use the PMI features between a slot of a case frame and its antecedent candidate to reflect such preferences:

e.g.
- \( \log \{ P(\text{child}|\text{cf}=\text{yaku}(2),\text{case}=\text{ni}) / P(\text{child}) \} \)

As well as most other features, this type of features is distinguished by the case of zero pronouns: ‘ga,’ ‘wo,’ and ‘ni,’ respectively.

Cluster, Category, and NE PMI We also use generalized example versions of word PMI to alleviate the data sparseness problem in word PMI:

e.g.
- \( \log \{ P([\text{cl}:32] | \text{yaku}(2), \text{ni}) / P([\text{cl}:32]) \} \)
- \( \log \{ P([\text{ct}:\text{PERSON}] | \text{yaku}(2), \text{ni}) / P([\text{ct}:\text{PERSON}]) \} \)
- \( \log \{ P([\text{ne}:\text{PERSON}] | \text{yaku}(2), \text{ni}) / P([\text{ne}:\text{PERSON}]) \} \)

After this, we will generically call these three features GE PMI. All the features described above overlap, and only their granularities are different.

Occupancy of case slot We believe that there is a relation between the occupancy of a case slot and its generativity of zero pronouns. For example, since the ni (dative) case of ‘yaku (3)’ often appears, we assume this slot is just omitted as a zero pronoun even if there is no overt argument. However, since the ni (dative) case of ‘yaku (1)’ rarely appears, we assume there is no case slot if there is no overt argument.

Therefore, we use the log occupancy of the case slot, whose value is the same as the log of the generative probability of a case slot in Kawahara and Kurohashi (2006)’s model and estimated from case structure analysis of a large raw corpus:

e.g.
- \( \log P(A(cs)|\text{cf}=\text{yaku}(2),\text{case}=\text{ni}) \)
  where \( A(cs) = 1 \) denotes that the target case slot is occupied by an overt argument.

Overt argument assignment score Our model not only takes into account the correspondence between a case slot and its omitted argument, but also the correspondence between a case slot and its overt argument. The score for overt argument assignment reflects the likelihood of an overt argument assignment, and this is the same as the score for the predicate-argument structure in Kawahara and Kurohashi (2006)’s model, which does not take into consideration zero anaphoric relations.

e.g.
- \( \log P(\text{cf}=\text{yaku}(2), gaw_o;\text{watashi}, w_o;\text{te}|\text{yaku}) \)

Only this feature is not separately calculated for each case, but for the whole overt predicate-argument structure. Note that, this feature also includes non-lexical preferences in the analysis of a predicate-argument structure.

5.2 Non-lexical features

We also exploit three types of non-lexical features, which are binary features to reflect syntactic and
Intra-sentential (64 categories)

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic</td>
<td>Mentioned with topic marker</td>
</tr>
<tr>
<td>IP-self</td>
<td>Mentioned at parent node</td>
</tr>
<tr>
<td>IC-self</td>
<td>Mentioned at child node</td>
</tr>
<tr>
<td>IGP-self</td>
<td>Mentioned at grand-parent node</td>
</tr>
<tr>
<td>IGC-self</td>
<td>Mentioned at grand-child node</td>
</tr>
<tr>
<td>IB-self</td>
<td>Mentioned at a preceding node in the sentence except above (before)</td>
</tr>
<tr>
<td>IA-self</td>
<td>Mentioned at a following node in the sentence except above (after)</td>
</tr>
<tr>
<td>IP-ga-ov</td>
<td>Overt nominative argument of a predicate in parent node</td>
</tr>
<tr>
<td>IP-ga-om</td>
<td>Omitted nominative argument of a predicate in parent node</td>
</tr>
<tr>
<td>IP-wo-ov</td>
<td>Overt accusative argument of a predicate in parent node</td>
</tr>
<tr>
<td>IP-wo-om</td>
<td>Omitted accusative argument of a predicate in parent node</td>
</tr>
<tr>
<td>IGP-ga-ov</td>
<td>Overt nominative argument of a predicate in grand-parent node</td>
</tr>
</tbody>
</table>

Inter-sentential (21 categories)

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>B1</td>
<td>Mentioned in the adjacent sentence</td>
</tr>
<tr>
<td>B1-ga-ov</td>
<td>Overt nominative argument of a predicate in the adjacent sentence</td>
</tr>
<tr>
<td>B1-ga-om</td>
<td>Omitted nominative argument of a predicate in the adjacent sentence</td>
</tr>
<tr>
<td>B1-wo-ov</td>
<td>Overt accusative argument of a predicate in the adjacent sentence</td>
</tr>
<tr>
<td>B2</td>
<td>Mentioned in two sentences before</td>
</tr>
<tr>
<td>B2-ga-ov</td>
<td>Overt nominative argument of a predicate in the two sentence before</td>
</tr>
<tr>
<td>B3</td>
<td>Mentioned in more than two sentences before</td>
</tr>
</tbody>
</table>

Table 4: Examples of case/location categories.

discourse preferences. Since Web text slightly adheres to formal grammar and thus rich syntactic preferences are not considered very important for resolution of zero anaphora in Web text, we only introduce simple features as non-lexical features.

Case/location We use case/location features to reflect syntactic, functional, and locational preferences. We considered 85 case/location categories, examples of which are summarized in Table 4. If an antecedent candidate appears in a certain case/location category, the corresponding feature value is 1; otherwise 0. These features are made for each case, respectively, i.e. there are a total of 255 case/location features.

Salience Previous work has reported the usefulness of salience in anaphora resolution (Lappin and Leass, 1994; Mitkov et al., 2002; Sasano et al., 2008). We introduce salience features to take into account the salience of each discourse entity. First, we apply the following simple rules to estimating the salience of each entity, and we then set salience features, whose value is 1 if the salience of an antecedent candidate is no less than 1.0; otherwise 0.

- +2: mentioned with topical marker “wa.”
- +1: mentioned without topical marker “wa.”
- ×0.5: beginning of each sentence.

Case assigned features We introduce case assigned features for each case type. The value is 1 if the corresponding zero pronoun is assigned to an antecedent; otherwise 0.

If the weights for these features become larger, corresponding zero pronouns are assigned to antecedents more often. Thus, the weights for these features are regarded as parameters to control the recall/precision trade-off. Although we mainly evaluate our system by using the F-measure, the algorithm mentioned in Section 4.3 does not select a parameter that maximizes the F-measure. Thus, after parameters are estimated by the algorithm, we adjust the weights for these features to maximize the F-measure by using training or development data.

6 Experiments

6.1 Setting

We used the same data set as described in (Sasano et al., 2009). This data set consisted of 186 Web documents (979 sentences, 19,677 morphemes), in which all predicate-argument relations were manually annotated. There were 2,137 predicates in this corpus, and 683 zero anaphoric relations were annotated. We call this data set a Web Corpus after this. We performed 6-fold cross-validation. We used correct morphemes, named entities, dependency structures, and coreference relations that were manually annotated to concentrate on zero anaphora resolution. We tested 10 initial values as parameter Λ. Since the parameters converged to almost the same value for each initial value, we considered that our model achieved the global maximum parameters in most cases.

We applied two baseline models for comparison. The first was the model proposed by Sasano et al. (2008), which did not use a log-linear framework but exploited almost the same clues. In addition, we also conducted an experiment with merged case frames to verify the usefulness of the
Recall Precision F-measure
Sasano et al. (2008)'s model 0.341 0.306 0.322 (233/683) (233/762)
Proposed model with merged case frames 0.334 0.412 0.369 (228/683) (228/753)
Proposed model 0.379 0.403 0.391 (259/683) (259/642)

Table 5: Experimental results of zero anaphora resolution on Web Corpus.

<table>
<thead>
<tr>
<th>Case Type</th>
<th>Recall</th>
<th>Precision</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOM</td>
<td>0.504</td>
<td>0.460</td>
<td>0.481</td>
</tr>
<tr>
<td></td>
<td>(120/238)</td>
<td>(120/261)</td>
<td></td>
</tr>
<tr>
<td>Intra-sentential</td>
<td>0.460</td>
<td>0.387</td>
<td>0.420</td>
</tr>
<tr>
<td>Inter-sentential</td>
<td>0.400</td>
<td>0.375</td>
<td>0.387</td>
</tr>
<tr>
<td>ACC</td>
<td>0.250</td>
<td>0.474</td>
<td>0.321</td>
</tr>
<tr>
<td></td>
<td>(17/68)</td>
<td>(17/38)</td>
<td></td>
</tr>
<tr>
<td>Intra-sentential</td>
<td>0.163</td>
<td>0.194</td>
<td>0.177</td>
</tr>
<tr>
<td>Inter-sentential</td>
<td>0.743</td>
<td>0.736</td>
<td>0.753</td>
</tr>
<tr>
<td>DAT</td>
<td>0.105</td>
<td>0.376</td>
<td>0.158</td>
</tr>
<tr>
<td></td>
<td>(6/57)</td>
<td>(6/19)</td>
<td></td>
</tr>
<tr>
<td>Intra-sentential</td>
<td>0.098</td>
<td>0.263</td>
<td>0.143</td>
</tr>
<tr>
<td>Inter-sentential</td>
<td>0.551</td>
<td>0.519</td>
<td>0.532</td>
</tr>
</tbody>
</table>

Table 6: Detailed experimental results for the proposed model on the Web Corpus.

6.2 Experimental Results

Table 5 summarizes the experimental results of zero anaphora resolution on the Web Corpus. The results indicate that our proposed model outperformed both Sasano et al. (2008)'s model and the model with merged case frames, which demonstrates the effectiveness of the log-linear framework and the usefulness of the case frames that were constructed for each meaning of each verb/adjective.

Table 6 shows the performance of the proposed model for each case and for each of intra- and inter-sentential zero anaphoric relations. Since the Web Corpus consists of relatively short sentences, there are many inter-sentential zero anaphora. Compared with previous work (Taira et al., 2008; Imamura et al., 2009), our model can resolve inter-sentential zero anaphora in the Web Corpus with comparatively good performance.

6.3 Contribution of features

We eliminated feature types one by one to investigate the contribution each made. Table 7 presents the eliminated feature types and the performance without each type. This table indicates the importance of word PMI and case/location features, since we obtained 0.021 and 0.092 lower F-measures without these features, respectively.

On the other hand, generalized example versions of word PMI did not affect performance much. However, when all generalized example PMIs were eliminated, performance worsened. Therefore, we considered that cluster PMI, category PMI, and NE PMI could be clues for zero anaphora resolution, and confirmed that zero anaphora resolution could benefit from overlap-
Table 8: Weights of lexical features. The Bold font indicates that value of weight is larger than average of weights in the same row.

Table 9: Experimental results on the NAIST Text Corpus. R, P, and F denote recall, precision, and F-measure, respectively.

We conjectured that this was because the NAIST Text Corpus consisted of newspaper articles that included relatively long sentences with formal grammar, and thus rich syntactic patterns were quite effective. Our model also took into account syntactic clues by using case/location features. However, since we basically focused on the zero anaphora in Web text that only adhered slightly to formal grammar, we did not give priority to exploring effective syntactic patterns.

7 Conclusion

This paper presented a discriminative model for Japanese zero anaphora resolution that can exploit large-scale lexicalized case frames, as well as non-lexical features obtained from a relatively small annotated corpus. Experimental results on a Web text revealed that our model could effectively resolve zero anaphora. We plan to investigate new features for zero anaphora resolution including richer syntactic patterns and global constraints in future work.
References


