Chinese Zero Pronoun Resolution: A Joint Unsupervised Discourse-Aware Model Rivaling State-of-the-Art Resolvers

Chen Chen and **Vincent Ng**
Human Language Technology Research Institute
University of Texas at Dallas
Richardson, TX 75083-0688
{yzcchen,vince}@hlt.utdallas.edu

Abstract

We propose an unsupervised probabilistic model for zero pronoun resolution. To our knowledge, this is the first such model that (1) is trained on zero pronouns in an unsupervised manner; (2) jointly identifies and resolves anaphoric zero pronouns; and (3) exploits discourse information provided by a salience model. Experiments demonstrate that our unsupervised model significantly outperforms its state-of-the-art unsupervised counterpart when resolving the Chinese zero pronouns in the OntoNotes corpus.

1 Introduction

A zero pronoun (ZP) is a gap in a sentence that is found when a phonetically null form is used to refer to a real-world entity. An anaphoric zero pronoun (AZP) is a ZP that corefers with one or more preceding mentions in the associated text. Below is an example taken from the Chinese TreeBank (CTB), where the ZP (denoted as *pro*) refers to 俄罗斯 (Russia).

[俄罗斯] 作为米洛舍夫维奇一贯的支持者，*pro* 曾经提出调停这场政治危机。

([Russia] is a consistent supporter of Milošević, *pro* has proposed to mediate the political crisis.)

As we can see, ZPs lack grammatical attributes that are useful for overt pronoun resolution such as NUMBER and GENDER. This makes ZP resolution more challenging than overt pronoun resolution.

Automatic ZP resolution is typically composed of two steps. The first step, AZP identification, involves extracting ZPs that are anaphoric. The second step, AZP resolution, aims to identify an antecedent of an AZP. State-of-the-art ZP resolvers have tackled both of these steps in a supervised manner, training one classifier for AZP identification and another for AZP resolution (e.g., Zhao and Ng (2007), Kong and Zhou (2010)).

More recently, we have proposed an unsupervised AZP resolution model (henceforth the CN14 model) that rivals its supervised counterparts in performance (Chen and Ng, 2014). The idea is to resolve AZPs by using a probabilistic pronoun resolution model trained on *overt* pronouns in an *unsupervised* manner. This is an appealing approach, as its language-independent generative process enables it to be applied to languages where data annotated with ZP links are not available.

In light of the advantages of unsupervised models, we examine in this paper the possibility of advancing the state of the art in unsupervised AZP resolution. The design of our unsupervised model is motivated by a key question: can we resolve AZPs by using a probabilistic model trained on *zero* pronouns in an *unsupervised* manner? As mentioned above, the CN14 model was trained on overt pronouns, but it is not clear how much this helped its resolution performance. In particular, the contexts in which overt and zero pronouns occur may not statistically resemble each other. For example, a ZP is likely to be closer to its antecedent than its overt counterpart. As another example, the verbs governing a ZP and its antecedent are more likely to be identical than the verbs governing an overt pronoun and its antecedent. Given such differences, it is not clear whether the knowledge learned from overt pronouns is always applicable to the resolution of AZPs. For this reason, we propose to train an *unsupervised* AZP resolution model directly on *zero* pronouns. Moreover, while we previously employed a *pipeline* architecture where we (1) used a set of heuristic rules for AZP identification, and then (2) applied their probabilistic model to all and only those ZPs that were determined to be anaphoric (Chen and Ng, 2014), in this work we identify and resolve AZPs in a *joint* fashion. To our knowledge, the model we are

Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers), pages 320–326,
Beijing, China, July 26-31, 2015. ©2015 Association for Computational Linguistics

proposing here is the first unsupervised model for joint AZP identification and resolution.[1]

In addition, motivated by work on *overt* pronoun resolution, we hypothesize that AZP resolution can be improved by exploiting *discourse* information. Specifically, we design a model of salience and incorporate salience information into our model as a feature. Inspired by traditional work on discourse-based anaphora resolution (e.g., Lappin and Leass (1994)), we compute salience based on the coreference clusters constructed so far using a rule-based coreference resolver. While ZPs have been exploited to improve coreference resolution (Kong and Ng, 2013), we are the first to improve AZP resolution using coreference information.

When evaluated on the Chinese portion of the OntoNotes corpus, our AZP resolver outperforms the CN14 model, achieving state-of-the-art results.

2 Related Work

Early approaches to AZP resolution employed *heuristic* rules to resolve AZPs in Chinese (e.g., Converse (2006), Yeh and Chen (2007)) and Spanish (e.g., Ferrández and Peral (2000)). More recently, *supervised* approaches have been extensively employed to resolve AZPs in Chinese (e.g., Zhao and Ng (2007), Kong and Zhou (2010), Chen and Ng (2013)), Korean (e.g., Han (2006)), Japanese (e.g., Seki et al. (2002), Isozaki and Hirao (2003), Iida et al. (2003; 2006; 2007), Imamura et al. (2009), Iida and Poesio (2011), Sasano and Kurohashi (2011)), and Italian (e.g., Iida and Poesio (2011)). As mentioned before, in order to reduce reliance on annotated data, we recently proposed an *unsupervised* probabilistic model for Chinese AZP resolution that rivaled its supervised counterparts in performance (Chen and Ng, 2014).

3 The Generative Model

Next, we present our model for jointly identifying and resolving AZPs in an unsupervised manner.

3.1 Notation

Let z be a ZP. C, the set of candidate antecedents of z, contains (1) the maximal or modifier NPs that precede z in the associated text that are at most two sentences away from it; and (2) a dummy candidate antecedent d (to which z will be resolved if it is non-anaphoric). k is the context surrounding z as well as every candidate antecedent c in C; k_c is the context surrounding z and candidate antecedent c; and l is a binary variable indicating whether c is the correct antecedent of z.

3.2 Training

Our model estimates $P(z, k, c, l)$, the probability of seeing (1) the ZP z; (2) the context k surrounding z and its candidate antecedents; (3) a candidate antecedent c of z; and (4) whether c is the correct antecedent of z. Since we estimate this probability from a raw, unannotated corpus, we are treating z, k, and c as observed data[2] and l as hidden data.

Motivated in part by previous work on English overt pronoun resolution (e.g., Cherry and Bergsma (2005) and Charniak and Elsner (2009)), we estimate the model parameters using the Expectation-Maximization algorithm (Dempster et al., 1977). Specifically, we use EM to iteratively (1) estimate the model parameters from data in which each ZP is labeled with the probability that it corefers with each of its candidate antecedents, and (2) apply the resulting model to re-label each ZP with the probability that it corefers with each of its candidate antecedents. Below we describe the details of the E-step and the M-step.

3.2.1 E-Step

The goal of the E-step is to compute $P(l{=}1|z, k, c)$, the probability that a candidate antecedent c is the correct antecedent of z given context k. Applying the definition of conditional probability and the Theorem of Total Probability, we can rewrite $P(l{=}1|z, k, c)$ as follows:

$$P(l{=}1|z, k, c) = \frac{P(z, k, c, l{=}1)}{P(z, k, c, l{=}1) + P(z, k, c, l{=}0)} \quad (1)$$

Assuming that exactly one of z's candidate antecedents is its correct antecedent, we can rewrite $P(z, k, c, l{=}0)$ as follows:

$$P(z, k, c, l{=}0) = \sum_{c' \in C, c' \neq c} P(z, k, c', l{=}1) \quad (2)$$

Given Equation (2), we can rewrite

[1] Note that Iida and Poesio (2011) perform joint *inference* over an AZP identification model and an AZP resolution model trained separately, not joint *learning* of the two tasks.

[2] Here, we treat z as observed data because we assume that the set of ZPs has been identified by a separate process. We adopt the heuristics for ZP identification that we introduced in Chen and Ng (2014).

$P(l{=}1|z,k,c)$ as follows:

$$P(l{=}1|z,k,c) = \frac{P(z,k,c,l{=}1)}{\sum_{c' \in C} P(z,k,c',l{=}1)} \quad (3)$$

Applying the Chain Rule, we can rewrite $P(z,k,c,l{=}1)$ as follows:

$$P(z,k,c,l{=}1) = P(z|k,c,l{=}1) * P(l{=}1|k,c) * P(c|k) * P(k) \quad (4)$$

Next, since z is a phonetically null form (and therefore is not represented by any linguistic attributes), we assume that each of its candidate antecedents and the associated context has the same probability of generating it. So we can rewrite $P(z|k,c,l{=}1)$ as follows:

$$P(z|k,c,l{=}1) = P(z|k,c',l{=}1) \;\; \forall\, c,c' \in C \quad (5)$$

Moreover, we assume that (1) given z and c's context, the probability of c being the antecedent of z is not affected by the context of the other candidate antecedents; and (2) k_c is sufficient for determining whether c is the antecedent of z. So,

$$P(l{=}1|k,c) \approx P(l{=}1|k_c,c) \approx P(l{=}1|k_c) \quad (6)$$

Next, applying Bayes Rule to $P(l{=}1|k_c)$, we get:

$$\frac{P(k_c|l{=}1)P(l{=}1)}{P(k_c|l{=}1)P(l{=}1) + P(k_c|l{=}0)P(l{=}0)} \quad (7)$$

Representing k_c as a set of n features $f_c^1, \ldots f_c^n$ and assuming that each f_c^i is conditionally independent given l, we can approximate Expression (7) as:

$$\frac{\prod_i P(f_c^i|l{=}1)P(l{=}1)}{\prod_i P(f_c^i|l{=}1)P(l{=}1) + \prod_i P(f_c^i|l{=}0)P(l{=}0)} \quad (8)$$

Furthermore, we assume that given context k, each candidate antecedent of z is generated with equal probability. In other words,

$$P(c|k) = P(c'|k) \;\; \forall\, c,c' \in C \quad (9)$$

Given Equations (4), (5), (8) and (9), we can rewrite $P(l{=}1|z,k,c)$ as:

$$\begin{aligned} P(l{=}1|z,k,c) &= \frac{P(z,k,c,l{=}1)}{\sum_{c' \in C} P(z,k,c',l{=}1)} \\ &= \frac{P(z|k,c,l{=}1)*P(l{=}1|k,c)*P(c|k)}{\sum_{c' \in C} P(z|k,c',l{=}1)*P(l{=}1|k,c')*P(c'|k)} \\ &\approx \frac{P(l{=}1|k_c)}{\sum_{c' \in C} P(l{=}1|k_{c'})} \approx \frac{\frac{\prod_i P(f_c^i|l=1)}{Z_c}}{\sum_{c' \in C} \frac{\prod_i P(f_{c'}^i|l=1)}{Z_{c'}}} \end{aligned} \quad (10)$$

where

$$Z_x = \prod_i P(f_x^i|l{=}1)P(l{=}1) + \prod_i P(f_x^i|l{=}0)P(l{=}0) \quad (11)$$

As we can see from Equation (10), our model has one group of parameters, namely $P(f_c^i|l{=}1)$. Using Equation (10) and the current parameter estimates, we can compute $P(l{=}1|z,k,c)$.

A point deserves mention before we describe the M-step. By including d as a dummy candidate antecedent for each z, we effectively model AZP identification and resolution in a joint fashion. If the model resolves z to d, it means that the model posits z as non-anaphoric; on the other hand, if the model resolves z to a non-dummy candidate antecedent c, it means that the model posits z as anaphoric and c as z's correct antecedent.

3.2.2 M-Step

Given $P(l{=}1|z,k,c)$, the goal of the M-step is to (re)estimate the model parameters, $P(l{=}1|k_c)$, using maximum likelihood estimation. Specifically, $P(l{=}1|k_c)$ is estimated as follows:

$$P(l{=}1|k_c) = \frac{Count(k_c, l{=}1) + \theta}{Count(k_c) + \theta * 2} \quad (12)$$

where $Count(k_c)$ is the number of times k_c appears in the training data, $Count(k_c, l{=}1)$ is the expected number of times k_c is the context surrounding an AZP and its antecedent c, and θ is the Laplace smoothing parameter, which we set to 1. Given context k'_c, we compute $Count(k'_c, l{=}1)$ as follows:

$$Count(k'_c, l{=}1) = \sum_{k: k_c = k'_c} P(l{=}1|z,k,c) \quad (13)$$

To start the induction process, we initialize all parameters with uniform values. Specifically, $P(l{=}1|k_c)$ is set to 0.5. Then we iteratively run the E-step and the M-step until convergence.

There is an important question we have not addressed: what features should we use to represent context k_c, which we need to estimate $P(l{=}1|k_c)$? We answer this question in Section 4.

3.3 Inference

After training, we can apply the resulting model to resolve ZPs. Given a test document, we process its ZPs in a left-to-right manner. For each ZP z enountered, we determine its antecedent as follows:

$$\hat{c} = \arg\max_{c \in C} P(l{=}1|z,k,c) \quad (14)$$

where C is the set of candidate antecedents of z. If we resolve a ZP to a preceding NP c, we fill its gap with c. Hence, when we process a ZP z, all of its preceding AZPs in the associated text have already been resolved, having had their gaps filled with their associated NPs. To resolve z, we create test instances between z and its candidate antecedents in the same way we described before. The only difference is that z's candidate antecedents may now include the NPs to which previous AZPs were resolved. In other words, this incremental resolution procedure may increase the number of candidate antecedents of each ZP z. Some of these additional candidate antecedents are closer to z than were their parent NPs, thus facilitating the resolution of z to the NPs in the following way: If the model resolves z to the additional candidate antecedent that fills the gap left behind by, say, AZP z', we postprocess the output by resolving z to the NP that z' is resolved to.[3]

4 Context Features

To fully specify our model, we need to describe how to represent k_c, which is needed to compute $P(l{=}1|k_c)$. Recall that k_c encodes the context surrounding candidate antecedent c and the associated ZP z. As described below, we represent k_c using eight features. Note that (1) all but feature 1 are computed based on syntactic parse trees, and (2) features 2, 3, and 6 are ternary-valued features.

1. the sentence distance between c and z;
2. whether the node spanning c has an ancestor NP node; if so, whether this NP node is a descendant of c's lowest ancestor IP node;
3. whether the node spanning c has an ancestor VP node; if so, whether this VP node is a descendant of c's lowest ancestor IP node;
4. whether vp has an ancestor NP node, where vp is the VP node spanning the VP that follows z;
5. whether vp has an ancestor VP node;
6. whether z is the first word of a sentence; if not, whether z is the first word of an IP clause;
7. whether c is a subject whose governing verb is lexically identical to the verb governing z;
8. c's salience rank (see Section 5).

	Training	Test
Documents	1,391	172
Sentences	36,487	6,083
Words	756,063	110,034
AZPs	–	1,713

Table 1: Statistics on the training and test sets.

Note that features 1, 2, 3 and 7 are not directly applicable to the dummy candidate. To compute the feature values of the dummy candidate, we first find the highest ranking non-dummy entity E in the salience list, and then set the values of these four features of the dummy candidate to the corresponding feature values of the rightmost mention of E. The motivation is that we want the dummy candidate to compete with the most salient non-dummy candidate.

5 Adding Salience

Recall from Section 4 that feature 8 requires the computation of salience. Intuitively, salient entities are more likely to contain the antecedent of an AZP.

We model salience as follows. For each ZP z, we compute the salience score for each (partial) entity preceding z.[4] To reduce the size of the list of preceding entities, we only consider a partial entity *active* if one of its mentions appears within two sentences of the active ZP z. We compute the salience score of each active entity w.r.t. z using the following equation:

$$\sum_{m \in E} g(m) * decay(m) \quad (15)$$

where m is a mention belonging to active entity E, $g(m)$ is a grammatical score which is set to 4, 2, or 1 depending on whether m's grammatical role is SUBJECT, OBJECT, or OTHER, respectively, and $decay(m)$ is decay factor that is set to 0.5^{dis} (where dis is the sentence distance between m and z). After computing the scores, we first sort the list of the active entities in descending order of salience. Then, within each active entity, we sort the mentions in increasing order of distance from z. Finally, we set the salience rank of each mention m to its position in the sorted list, but cap the rank

[3] This postprocessing step is needed because the additional candidate antecedents are only gap fillers.

[4] We compute the list of preceding entities automatically using SinoCoreferencer, a publicly available Chinese entity coreference resolver. See `http://www.hlt.utdallas.edu/~yzcchen/coreference/`.

	Setting 1: Gold Parses, Gold AZPs						Setting 2: Gold Parses, System AZPs						Setting 3: System Parses, System AZPs					
	Baseline			Our Model			Baseline			Our Model			Baseline			Our Model		
Source	R	P	F	R	P	F	R	P	F	R	P	F	R	P	F	R	P	F
Overall	47.5	47.9	47.7	50.0	50.4	50.2	35.4	21.0	26.4	35.7	26.2	30.3	19.9	12.9	15.7	19.6	15.5	17.3
NW	41.7	41.7	41.7	46.4	46.4	46.4	29.8	24.8	27.0	32.1	28.1	30.0	11.9	13.0	12.4	11.9	14.3	13.0
MZ	34.0	34.2	34.1	38.9	39.1	39.0	24.1	14.5	18.1	29.6	19.6	23.6	6.2	5.2	5.7	4.9	4.7	4.8
WB	47.9	47.9	47.9	51.8	51.8	51.8	37.3	18.7	24.9	39.1	22.9	28.9	19.0	11.3	14.2	20.1	14.3	16.7
BN	52.8	52.8	52.8	53.8	53.8	53.8	31.5	28.1	29.7	30.8	30.7	30.7	18.2	19.5	18.8	18.2	22.3	20.0
BC	49.8	50.3	50.0	49.2	49.6	49.4	38.0	21.0	27.0	35.9	26.6	30.6	20.6	12.4	15.5	19.4	14.6	16.7
TC	45.2	46.7	46.0	51.9	53.5	52.7	42.4	20.3	27.4	43.5	28.7	34.6	32.2	13.3	18.8	31.8	17.0	22.2

Table 2: AZP resolution results of the baseline and our model on the test set.

at 5 in order to reduce sparseness during parameter estimation.

Note that the above list contains only non-dummy entities. We model the salience of a dummy entity D, which contains only the dummy candidate for z, as follows. Intuitively, if z is non-anaphoric, D should be the most salient entity. Hence, we put D at the top of the list if z satisfies any of the following three conditions, all of which are strong indicators of non-anaphoricity: (1) z appears at the beginning of a document; (2) the verb following z is 有 (there is) or 没有 (there is not) with part of speech VE; or (3) the VP node in the syntactic parse tree following z does not span any verb. If none of these conditions is satisfied, we put D at the bottom of the list.

6 Evaluation

6.1 Experimental Setup

Datasets. We employ the Chinese portion of the OntoNotes 5.0 corpus that was used in the official CoNLL-2012 shared task (Pradhan et al., 2012). In the CoNLL-2012 data, the training set and development set contain ZP coreference annotations, but the test set does not. Therefore, we train our models on the training set and perform evaluation on the development set. Statistics on the datasets are shown in Table 1. The documents in these datasets come from six sources, namely Broadcast News (BN), Newswires (NW), Broadcast Conversations (BC), Telephone Conversations (TC), Web Blogs (WB), and Magazines (MZ).

Evaluation measures. We express results in terms of recall (R), precision (P), and F-score (F) on resolving AZPs, considering an AZP z correctly resolved if it is resolved to any NP in the same coreference chain as z.

Evaluation settings. Following Chen and Ng (2014), we evaluate our model in three settings. In Setting 1, we assume the availability of gold syntactic parse trees and gold AZPs.[5] In Setting 2, we employ gold syntactic parse trees and system (i.e., automatically identified) AZPs. Finally, in Setting 3 (the end-to-end setting), we employ system syntactic parse trees and system AZPs. The gold and system syntactic parse trees, as well as the gold AZPs, are obtained from the CoNLL-2012 shared task dataset, while the system AZPs are identified by our generative model.

6.2 Results

As our baseline, we employ the CN14 system, which has achieved the best result to date on our test set. Table 2 shows results obtained using both the baseline system and our model on the entire test set as well as on each of the six sources. As we can see, our model significantly[6] outperforms the baseline under all three settings by 2.5%, 3.9% and 1.6% respectively in terms of overall F-score.

7 Conclusion

We proposed a novel unsupervised model for Chinese zero pronoun resolution by (1) training on zero pronouns; (2) jointly identifying and resolving anaphoric zero pronouns; and (3) exploiting salience information. Experiments on the OntoNotes 5.0 corpus showed that our unsupervised model achieved state-of-the-art results.

Acknowledgments

We thank the three anonymous reviewers for their detailed and insightful comments on an earlier draft of this paper. This work was supported in part by NSF Grants IIS-1147644 and IIS-1219142. Any opinions, findings, conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views or official policies, either expressed or implied, of NSF.

[5] When gold AZPs are used (i.e., Setting 1), we simply remove the dummy candidate antecedent from the list of candidate antecedents during inference.

[6] All significance tests are paired t-test, with $p < 0.05$.

References

Eugene Charniak and Micha Elsner. 2009. EM works for pronoun anaphora resolution. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 148--156.

Chen Chen and Vincent Ng. 2013. Chinese zero pronoun resolution: Some recent advances. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1360--1365.

Chen Chen and Vincent Ng. 2014. Chinese zero pronoun resolution: An unsupervised probabilistic model rivaling supervised resolvers. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 763--774.

Colin Cherry and Shane Bergsma. 2005. An expectation maximization approach to pronoun resolution. In *Proceedings of the Ninth Conference on Natural Language Learning*, pages 88--95.

Susan Converse. 2006. *Pronominal Anaphora Resolution in Chinese*. Ph.D. thesis, University of Pennsylvania.

Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39:1--38.

Antonio Ferrández and Jesús Peral. 2000. A computational approach to zero-pronouns in Spanish. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 166--172.

Barbara J. Grosz, Aravind K. Joshi, and Scott Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203--226.

Na-Rae Han. 2006. *Korean zero pronouns: analysis and resolution*. Ph.D. thesis, University of Pennsylvania.

Jerry Hobbs. 1978. Resolving pronoun references. *Lingua*, 44:311--338.

Ryu Iida and Massimo Poesio. 2011. A cross-lingual ILP solution to zero anaphora resolution. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 804--813.

Ryu Iida, Kentaro Inui, Hiroya Takamura, and Yuji Matsumoto. 2003. Incorporating contextual cues in trainable models for coreference resolution. In *Proceedings of the EACL Workshop on The Computational Treatment of Anaphora*, pages 23--30.

Ryu Iida, Kentaro Inui, and Yuji Matsumoto. 2006. Exploting syntactic patterns as clues in zero-anaphora resolution. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 625--632.

Ryu Iida, Kentaro Inui, and Yuji Matsumoto. 2007. Zero-anaphora resolution by learning rich syntactic pattern features. *ACM Transactions on Asian Language Information Processing*, 6(4).

Kenji Imamura, Kuniko Saito, and Tomoko Izumi. 2009. Discriminative approach to predicate-argument structure analysis with zero-anaphora resolution. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 85--88.

Hideki Isozaki and Tsutomu Hirao. 2003. Japanese zero pronoun resolution based on ranking rules and machine learning. In *Proceedings of the 2003 Conference on Empirical methods in natural language processing*, pages 184--191.

Fang Kong and Hwee Tou Ng. 2013. Exploiting zero pronouns to improve chinese coreference resolution. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 278--288.

Fang Kong and Guodong Zhou. 2010. A tree kernel-based unified framework for Chinese zero anaphora resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 882--891.

Shalom Lappin and Herbert Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535--562.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Proceedings of 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning: Shared Task*, pages 1--40.

Ryohei Sasano and Sadao Kurohashi. 2011. A discriminative approach to Japanese zero anaphora resolution with large-scale lexicalized case frames. In *Proceedings of the 5th International Joint Conference on Natural Language Processing*, pages 758--766.

Kazuhiro Seki, Atsushi Fujii, and Tetsuya Ishikawa. 2002. A probabilistic method for analyzing Japanese anaphora integrating zero pronoun detection and resolution. In *Proceedings of the 19th International Conference on Computational linguistics*.

Ching-Long Yeh and Yi-Chun Chen. 2007. Zero anaphora resolution in Chinese with shallow parsing. *Journal of Chinese Language and Computing*, 17(1):41--56.

Shanheng Zhao and Hwee Tou Ng. 2007. Identification and resolution of Chinese zero pronouns: A machine learning approach. In *Proceedings of the 2007 Joint Conference on Empirical Methods on Natural Language Processing and Computational Natural Language Learning*, pages 541--550.

GuoDong Zhou, Fang Kong, and Qiaoming Zhu. 2008. Context-sensitive convolution tree kernel for pronoun resolution. In *Proceedings of the 3rd International Joint Conference on Natural Language Processing*, pages 25--31.