Chinese Common Noun Phrase Resolution: An Unsupervised Probabilistic Model Rivaling Supervised 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

Pronoun resolution and common noun phrase resolution are the two most challenging subtasks of coreference resolution. While a lot of work has focused on pronoun resolution, common noun phrase resolution has almost always been tackled in the context of the larger coreference resolution task. In fact, to our knowledge, there has been no attempt to address Chinese common noun phrase resolution as a standalone task. In this paper, we propose a generative model for unsupervised Chinese common noun phrase resolution that not only allows easy incorporation of linguistic constraints on coreference but also performs joint resolution and anaphoricity determination. When evaluated on the Chinese portion of the OntoNotes 5.0 corpus, our model rivals its supervised counterpart in performance.

Introduction

Common noun phrase (NP) resolution is the task of identifying and resolving the anaphoric common NPs in a text. Consider the following example of common NP resolution taken from the Chinese Treebank (Xue et al. 2005):

[蓝欣无线电厂] 全年亏损１４００万元，[企业] 接近破产的边缘。
[The Lanxin Wireless Factory] lost 14 million dollars in one year, and [the company] is on the verge of bankruptcy.

In this example, there are two common NPs, 企业 (the company) and 破产的边缘 (bankruptcy), and two named entities, 蓝欣无线电厂 (The Lanxin Wireless Factory) and １４００万元 (14 million dollars). While 破产的边缘 does not have an antecedent, 企业 refers to 蓝欣无线电厂.

In a recent analysis of a state-of-the-art English coreference resolver, Stoyanov et al. (2009) report that pronoun resolution and common NP resolution remain the most challenging subtasks of coreference resolution. While a lot of work has focused specifically on pronoun resolution (Mitkov 2002), the same is not true for common NP resolution.

This paper examines the task of Chinese common NP resolution. While much research has been done on English coreference resolution, there has been relatively little work on the corresponding problem of Chinese coreference resolution. Nevertheless, like its English counterpart, common NP resolution is one of the most challenging subtasks of Chinese coreference resolution (Chen and Ng 2012). Virtually all recent work on Chinese common NP resolution has been addressed in the context of *supervised* Chinese coreference resolution. Hence, the state-of-the-art in Chinese common NP resolution can only be approximated from the outputs of state-of-the-art supervised Chinese coreference resolvers (e.g., Kong and Ng (2013), Björkelund and Kuhn (2014)).

Our contribution lies in the proposal of the first *unsupervised* model for Chinese common NP resolution that rivals its supervised counterpart in performance when evaluated on the Chinese portion of the OntoNotes 5.0 corpus. Its main advantage is that it does not require training data with manually resolved anaphoric common NPs. The fact that its underlying generative process is not language dependent enables it to be applied to languages where such annotated data is not readily available. We train our resolver on a raw, unannotated Chinese corpus using the Expectation-Maximization (EM) algorithm (Dempster, Laird, and Rubin 1977).

At first glance, it may seem that common NPs can be resolved together with pronouns using existing unsupervised models for pronoun resolution (e.g., Cherry and Bergsma (2005), Charniak and Elsner (2009)) after augmenting their feature sets with features that are useful for common NP resolution. However, there are at least two compelling reasons for developing unsupervised models specifically for common NP resolution. First, feature design is more difficult for generative models than for discriminative models, as the former cannot handle overlapping features. Hence, to facilitate feature design, it would be better to train separate models for pronoun resolution and common NP resolution. Second, and perhaps more importantly, *anaphoricity determination*, the task of determining whether an NP is anaphoric and hence needs to be resolved, is more challenging for common NPs than for pronouns. The reason is that while there exist lexical and syntactic cues that can be used to reliably identify pleonastic pronouns (Bergsma and Yarowsky 2011), the lack of such cues in common NPs makes the identification of anaphoric common NPs challenging even in a supervised manner, let alone in an unsupervised manner. Note that ignoring anaphoricity determination and having our model attempt to resolve every common NP is not a viable option, as only 18% of the Chinese common NPs in our evaluation corpus (OntoNotes 5.0) are anaphoric.

In light of the difficulty of designing a standalone system for determining the anaphoricity of common NPs, we perform *joint* anaphoricity determination and common NP resolution in our model. Motivated in part by Rahman and Ng (2009; 2011), we model these two tasks jointly by introducing a *dummy* candidate antecedent for each common NP to be resolved to. A key challenge is to design features for encoding these dummy candidate antecedents.

Related Work

Virtually all work on Chinese common NP resolution was conducted in the context of Chinese coreference resolution. Broadly, related work on Chinese coreference resolution has focused on either the full coreference task (as defined in the ACE evaluations and the CoNLL-2012 shared task on multilingual coreference resolution, for instance) or the subtasks of overt pronoun resolution and zero pronoun resolution. Below we present representative work on each of these tasks.

Full coreference resolution. Luo and Zitouni (2005) employ Chinese specific syntactic features for supervised coreference resolution. Wang and Ngai (2006) apply clustering to unsupervised Chinese coreference resolution, employing features commonly-used for English coreference resolution. Björkelund and Kuhn (2014) learn structured perceptrons for supervised coreference resolution with latent antecedents and non-local features, achieving the best Chinese coreference results reported to date on the OntoNotes 5.0 corpus.

Overt pronoun resolution. Song et al. (2008) employ syntactic features and word senses to resolve third-person Chinese pronouns. Kong and Zhou (2012) exploit tree kernels to resolve Chinese overt pronouns. Chen and Ng (2014a) adopt a bilingual approach, improving Chinese overt pronoun resolution by exploiting English coreference-annotated data as well as English gender and number word lists.

Zero pronoun resolution. Work on this task has adopted a variety of approaches, including rule-based approaches (e.g., Converse (2006), Yeh and Chen (2007)), supervised learning approaches (e.g., Zhao and Ng (2007), Kong and Zhou (2010), Chen and Ng (2013)), and unsupervised learning approaches (e.g., Chen and Ng (2014b; 2014c)).

The Generative Model

Notation

We begin by introducing the notation that we use in the rest of this paper. We denote n to be the current common NP to be resolved (henceforth the *active* common NP) and C to be the set of candidate antecedents of n in the associated text. Following Vieira and Poesio's (2000) seminal work on the resolution of English definite descriptions, we do not allow common NPs to be resolved to pronouns. Rather, we restrict the set of candidate antecedents of n to contain only the non-pronominal NPs that appear before n in the associated text as well as a dummy candidate antecedent d (to which n will be resolved if it is non-anaphoric). Also, we define k to be the context surrounding n as well as every candidate antecedent c in C, and k_c to be the context surrounding n and candidate antecedent c. Moreover, we define l to be a binary variable indicating whether c is the correct antecedent of n. Each NP has three grammatical attributes, namely NUMBER, GENDER and ANIMACY, which are collectively denoted by A. a represents a specific attribute in A. Finally, n_a and c_a denote the values of n and c with respect to a respectively, and n_h and c_h denote n and c's respective head words.[1]

Training

Our model estimates $P(n, k, c, l)$, the probability of seeing (1) the active common noun n; (2) the context k surrounding n and its candidate antecedents; (3) a candidate antecedent c of n; and (4) l, a binary value indicating whether c is n's correct antecedent. Since we estimate this probability from a raw, unannotated corpus, we are effectively treating n, k, and c as observed data and l as hidden data.

Owing to the presence of hidden data, we estimate the model parameters using the EM algorithm. Specifically, we use EM to iteratively estimate the parameters from data in which each common NP is labeled with the probability that it corefers with each of its candidate antecedents, and apply the resulting model to relabel each common NP 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.

E-step. The goal of the E-step is to compute $P(l{=}1|n, k, c)$, the probability that a candidate antecedent c is the correct antecedent of n given context k. Assuming that exactly one of the n's candidate antecedents is its correct antecedent, we can rewrite $P(l{=}1|n, k, c)$ as follows:

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

As we can see from Equation (1), to compute $P(l{=}1|n, k, c)$, we need to compute $P(n, k, c, l{=}1)$, which can be rewritten using Chain Rule:

$$\begin{aligned} P(n, k, c, l{=}1) = P(n|k, c, l{=}1) * P(l{=}1|k, c) \\ * P(c|k) * P(k) \end{aligned} \quad (2)$$

Next, given $l = 1$ (i.e., c is the antecedent of n), we assume that we can generate n from c without looking at the context. Using this assumption and approximating n and c by their head words, we can rewrite $P(n|k, c, l{=}1)$ as follows:

$$P(n|k, c, l{=}1) \approx P(n_h|c_h, l{=}1) \quad (3)$$

Moreover, we assume that (1) given n and c's context, the probability of c being the antecedent of n 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 n. So,

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

Given Equations (2), (3) and (4), we can rewrite $P(l{=}1|n, k, c)$ as follows:

$$\begin{aligned} P(l{=}1|n, k, c) &= \frac{P(n, k, c, l{=}1)}{\sum_{c' \in C} P(n, k, c', l{=}1)} \\ &\approx \frac{P(n_h|c_h, l{=}1) * P(l{=}1|k_c) * P(c|k)}{\sum_{c' \in C} P(n_h|c'_h, l{=}1) * P(l{=}1|k_{c'}) * P(c'|k)} \end{aligned} \quad (5)$$

[1] We use the rightmost word of an NP as its head word.

As we can see from Equation (5), our model has three groups of parameters, namely $P(n_h|c_h, l{=}1)$, $P(l{=}1|k_c)$ and $P(c|k)$. With these three groups of parameters, we can apply Equation (5) to efficiently compute $P(l{=}1|n, k, c)$.

Two points deserve mention before we describe our M-step. First, among the three groups of parameters, $P(n_h|c_h, l{=}1)$ and $P(l{=}1|k_c)$ are estimated in the M-step described below, while $P(c|k)$ is computed heuristically (see the next section for details). Intuitively, $P(c|k)$ is the prior probability of a candidate antecedent c given context k. The simplest way to model $P(c|k)$ is to assume that every candidate antecedent is equally likely given the context. In practice, however, some candidate antecedents are implausible given the context (e.g., those that are grammatically incompatible with the anaphor), and hence the model could be improved by not assigning any probability mass to linguistically implausible candidate antecedents. As we will see in the next section, we will identify such candidate antecedents based on a set of linguistic constraints on coreference.

Second, by including d as a dummy candidate antecedent for each n, we effectively model anaphoricity determination and common NP resolution in a joint fashion. If the model resolves n to d, it means that the model posits n as non-anaphoric; on the other hand, if the model resolves n to a non-dummy candidate antecedent c, it means that the model posits n as anaphoric and c as n's correct antecedent.

M-step. Given $P(l{=}1|n, k, c)$, the goal of the M-step is to (re)estimate two of the three groups of model parameters mentioned above, namely $P(n_h|c_h, l{=}1)$ and $P(l{=}1|k_c)$, using maximum likelihood estimation.

Specifically, $P(n_h|c_h, l{=}1)$ is estimated as follows:

$$P(n_h|c_h, l{=}1) = \frac{Count(n_h, c_h, l{=}1) + \theta}{Count(c_h, l{=}1) + \theta * |h|} \quad (6)$$

where $Count(c_h, l{=}1)$ is the expected number of times c has head word c_h when it is the antecedent of a common NP; and $|h|$ is the number of possible head words in the training data (note that we designate a unique symbol to be the "head word" of all dummy candidate antecedents). Also, θ is the Laplace smoothing parameter, which we set to 1, and $Count(n_h, c_h, l{=}1)$ is the expected number of times n has n_h as its head when its antecedent c has head c_h. Given head words n'_h and c'_h, we compute $Count(n'_h, c'_h, l{=}1)$ as follows:

$$Count(n'_h, c'_h, l{=}1) = \sum_{n,c:n_h=n'_h, c_h=c'_h} P(l{=}1|n, k, c) \quad (7)$$

Similarly, $P(l{=}1|k_c)$ is estimated as follows:

$$P(l{=}1|k_c) = \frac{Count(k_c, l{=}1) + \theta}{Count(k_c) + \theta * 2} \quad (8)$$

where $Count(k_c)$ is the number of times k_c appears in the training data, and $Count(k_c, l{=}1)$ is the expected number of times k_c is the context surrounding an active common NP and its antecedent c. Given context k'_c, we compute $Count(k'_c, l{=}1)$ as follows:

$$Count(k'_c, l{=}1) = \sum_{k:k_c=k'_c} P(l{=}1|n, k, c) \quad (9)$$

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

There are two important questions we have not addressed so far. First, as mentioned before, $P(c|k)$ is computed using a set of constraints that identify linguistically plausible candidate antecedents, but which constraints should we use? Second, what features should we use to represent context k_c, which we need to estimate $P(l{=}1|k_c)$? We defer the discussion of these questions to the subsequent sections.

Inference

After training, we can apply the resulting model to resolve common NPs. Given a common NP n, we determine its antecedent as follows:

$$\hat{c} = \arg\max_{c \in C} P(l{=}1|n, k, c) \quad (10)$$

where C is the set of candidate antecedents of n, as defined at the beginning of this section. In other words, we apply Equation (10) to each of n's candidate antecedents, and select the $c \in C$ that yields the largest probability. If c is a real (i.e., non-dummy) candidate antecedent, we posit c as the correct antecedent of n; otherwise, we posit n as non-anaphoric.

Priors on Candidate Antecedents

In this section, we show how to compute $P(c|k)$, the prior probability of a candidate antecedent c given context k. To do this, we use a set of linguistic constraints on coreference whose violation implies that a candidate antecedent cannot be coreferent with a common NP. Hence, if c violates one of these constraints, we set $P(c|k)$ to 0 and distribute the probability mass uniformly over those candidates that survive all of the constraints. Since these constraints are not applicable to dummy candidates, we assume for simplicity that they always survive this candidate filtering step. Below we describe these constraints and subsequently explain how to compute the information they require.

Constraints

As mentioned above, each constraint is applicable to an active common NP n and one of its candidate antecedents c.

Grammatical consistency. This constraint specifies that c and n have to be compatible with respect to three grammatical attributes, namely NUMBER, GENDER and ANIMACY. We will describe our method for determining the value of each of these attributes in the next subsection.

Semantic compatibility. This constraint specifies that c cannot be coreferent with n if replacing n with c's head violates the selectional restriction imposed by n's governing verb. Our approach to compute selectional restrictions resembles those of Kehler et al. (2004) and Yang, Su, and Tan (2005). Specifically, for each verb and each noun that serves as a subject or an object in the Chinese Gigaword corpus (Robert et al. 2009), we compute their mutual information (MI), and assume that a noun violates the selectional restriction imposed by a verb if their MI is less than zero.

i-within-i. The i-within-i constraint is a linguistic constraint that disallows coreference between two NPs if they have a parent-child relationship in the associated parse tree unless the child is an appositive.

Extra modifier. This constraint will be violated if c has one or more modifiers that do not appear in n's list of modifiers.[2] We treat a word in an NP as a modifier unless it is a stopword, a punctuation, or the NP's head word.

Attributes of Candidate Antecedents

Recall that the grammatical consistency constraint above requires the computation of three grammatical attribute values (NUMBER, GENDER, ANIMACY) of a candidate antecedent, which, as mentioned before, is a non-proniminal NP. While there exist publicly available English word lists that can be used to look up essential attributes of an English NP (Ji and Lin 2009), such resources are not available for Chinese. As a result, we need to define our methods for computing the values of these grammatical attributes.

ANIMACY. We determine the ANIMACY of a candidate antecedent c heuristically. Specifically, we first check the NP type of c. If c is a named entity, there are two cases to consider: if c is a *person*, we label it as *animate*; otherwise, we label it as *inanimate*.[3] If c is a common NP, we look up the ANIMACY of its head noun in an automatically constructed word list WL. If the head noun is not in WL, we set its ANIMACY to *unknown*.

Next, we describe our method for constructing WL, which is motivated by the observation that measure words are pervasively used to modify common nouns in Chinese. Specifically, some measure words are used to modify *inanimate* nouns only. For example, the nouns modified by the measure word 张 are always *inanimate*, as in 一张纸 (one piece of paper). On the other hand, some measure words are used to modify *animate* nouns only. For example, the nouns modified by the measure word 位 are always *animate*, as in 一位工人 (one worker).

Given this observation, we first define two lists, M_{ani} and M_{inani}. M_{ani} is a list of measure words that can only modify *animate* nouns. M_{inani} is a list of measure words that can only modify *inanimate* nouns.[4] There exists a special measure word, 个, which can be used to modify most of the common nouns regardless of their ANIMACY. As a result, we remove 个 from both lists. After constructing M_{ani} and M_{inani}, we (1) parse the Chinese Gigaword corpus using an efficient dependency parser, ctbparser[5] (Qian et al. 2010), and then (2) collect all pairs of words (m, n), where m is a measure word, n is a common noun, and there is a NMOD dependency relation between m and n. Finally, we determine the ANIMACY of a given common NP n as follows. First, we retrieve all of the pairs containing n. Then, we sum over all occurrences of m in M_{ani} (call the sum C_{ani}), as well as all occurrences of m in M_{inani} (call the sum C_{inani}). If $C_{ani} > C_{inani}$, we label this common NP as *animate*; otherwise, we label it as *inanimate*.

GENDER. We determine the GENDER of a candidate antecedent c based on its ANIMACY. Specifically, if c is *inanimate*, we set its GENDER to *neuter*. Otherwise, we determine its gender by looking it up in a gender word list we constructed using Bergsma and Lin's (2006) approach. If its head noun is not in the list, we set its GENDER to *masculine* by default.

Next, we describe how the gender word list is constructed. Following Bergsma and Lin (2006), we define a *dependency path* as the sequence of non-terminal nodes and dependency labels between two potentially coreferent entities in a dependency parse tree. From the parsed Chinese Gigaword corpus, we collect every dependency path that connects two pronouns. For each path P collected, we compute $CL(P)$, the coreference likelihood of P, as follows:

$$CL(P) = \frac{N_I(P)}{N_I(P) + N_D(P)} \tag{11}$$

where $N_I(P)$ is the number of times P connects two identical pronouns, and $N_D(P)$ is the number of times it connects two different pronouns. Assuming that two identical pronouns in a sentence are coreferent (Bergsma and Lin 2006), we can see that the larger a path's CL value is, the more likely it is that the two NPs it connects are coreferent. To ensure that we have dependency paths that are strongly indicative of coreference relations, we consider a dependency path P a *coreferent path* if and only if $CL(P) > 0.8$.

Given these coreferent paths, we determine the GENDER of a noun n as follows. We compute (1) $N_M(n)$, the number of coreferent paths connecting n with a *masculine* pronoun; and (2) $N_F(n)$, the number of coreferent paths connecting n with a *feminine* pronoun. If $N_F(n) > N_M(n)$, we set n's gender to *feminine*; otherwise, we set it to *masculine*.

NUMBER. When computing the NUMBER of a candidate antecedent c in English, Charniak and Elsner (2009) rely on part-of-speech information. For example, NN and NNP denote singular nouns, whereas NNS and NNPS denote plural nouns. However, Chinese part-of-speech tags do not provide such information. Hence, we need a different method for finding the NUMBER of a candidate antecedent c in Chinese. If c is a named entity, its NUMBER is *singular*. If c is a common NP, we infer its NUMBER from its string: if the string ends with 们 or is modified by a quantity word (e.g., 一些, 许多), c is *plural*; otherwise, c is *singular*.

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 active common NP n. As described below, we represent k_c using five features.

[2] There are exceptions to this rule, however. For instance, a document may first mention *this harsh winter* and then *the miserable cold*. These two NPs can be coreferent even though the second NP has an extra modifier. Nevertheless, since this phenomenon is rare, we decided to employ it as a constraint.

[3] A detailed description of our named entity recognizer can be found in Chen and Ng (2014d).

[4] We create these two lists with the help of this page: `http://chinesenotes.com/ref_measure_words.htm`

[5] `http://code.google.com/p/ctbparser/`

1. the logarithm of the sentence distance between c and n;[6]
2. a binary-valued feature indicating whether c and n have the same lexical string;
3. a four-valued feature indicating whether c_h and n_h are the same; if not, whether c_h starts with n_h; and if not, whether c_h ends with n_h;
4. a four-valued feature indicating one of the following possibilities: whether c and n have the same governing verb and the same grammatical role; whether c and n have the same governing verb but different grammatical roles; whether c and n have different governing verbs but the same grammatical role; whether c and n have different governing verbs and different grammatical roles;
5. a binary-valued feature indicating whether c is the highest-ranked candidate antecedent according to a simple candidate antecedent ranking strategy that favors recency as well as those candidates that satisfy some sort of relaxed head match condition. Specifically, the candidate antecedents are ranked as follows. First, we sort the non-dummy candidates so that those appearing later in the text are ranked higher than those appearing earlier in the text. Second, we rank the dummy candidate d above all of the non-dummy candidates. Finally, we rerank each non-dummy candidate c as follows: if c's head contains n's head, then we place c immediately above d.

Now that we can compute the aforementioned five features for a non-dummy candidate antecedent, we next specify how we compute these features for dummy candidate antecedent d of active mention n. To compute feature 1, we assign the sentence distance between n and d the value $n_{sid}+1$, where n_{sid} is the id of the sentence in which n appears. By doing so, we make the probability of picking d as the correct antecedent (i.e., the probability of classifying n as non-anaphoric) depend on the position in which n appears in the associated text. This makes sense because in general, the probability of n being non-anaphoric tends to be larger (smaller) when it appears earlier (later) in the text.

Features 2, 3 and 4 are computed as follows. We assume that (1) d and n do not have the same lexical string (for feature 2); (2) n_h is not the same as d_h and does not appear within d_h (for feature 3); and (3) d and n have different governing verbs and different grammatical roles (for feature 4).

Finally, feature 5 is computed by using the aforementioned ranking strategy, which has already taken into account d.

Evaluation

Experimental Setup

Datasets. We employ the Chinese portion of the OntoNotes 5.0 corpus that was used in the CoNLL-2012 shared task (Pradhan et al. 2012). The shared task organizers partitioned the documents in the corpus into a training set, a development set and a test set. We train our models on the combined training and development sets, and perform evaluation on the test set. Statistics on the datasets are shown in Table 1.[7] The documents in these datasets come from six sources, namely Broadcast News (BN), Newswire (NW), Broadcast Conversation (BC), Telephone Conversation (TC), Web Blog (WB) and Magazine (MZ).

	Training+Dev	Test
Documents	1,563	166
Sentences	42,570	4,472
Words	860K	90K
Anaphoric common NPs	–	3,268

Table 1: Statistics on the training+development and test sets.

Evaluation setting. Following the CoNLL-2012 shared task, we evaluate our system under the *end-to-end* setting, meaning that the common NPs to be resolved and the candidate antecedents are extracted automatically. The common NPs to be resolved are obtained by (1) extracting from the automatically computed syntactic parse trees (which are provided by the shared task organizers) all the maximal NPs, and (2) discarding those that are pronouns or named entities. The set of non-dummy candidate antecedents for a common NP is created by taking all the non-pronominal maximal NPs preceding the common NP in the associated document.

Evaluation measures. We express the results of common NP resolution in terms of recall (R), precision (P) and F-score (F), where R is the percentage of anaphoric common NPs that are correctly resolved, and P is the percentage of resolved common NPs that are anaphoric and correctly resolved.

Results

Baseline systems. We employ five resolvers as baseline systems. To gauge the difficulty of the task, we employ four simple rule-based resolvers, which resolve a common NP n to (1) the candidate antecedent closest to n (Baseline 1); (2) the subject candidate antecedent closest to n (Baseline 2); (3) the closest candidate antecedent with the same head as n (Baseline 3); and (4) the closest subject candidate antecedent with the same head as n (Baseline 4). These four baselines allow us to study the role of (1) recency, (2) salience, (3) recency combined with head match, and (4) salience combined with head match in common NP resolution respectively.

The remaining baseline is a supervised resolver that produces the best result to date on our test set (Björkelund and Kuhn 2014). Since this resolver outputs coreference chains, in order to compute recall and precision, we have to (1) identify from these response chains the resolved common NPs and (2) determine whether these common NPs are correctly resolved. We consider an NP in a response chain a resolved common NP if (1) it is neither a named entity nor a pronoun, (2) it is not the first element of the chain, and (3) at least one non-pronominal NP precedes it in the chain.[8] We assume

[6] We use the logarithm of the sentence distance rather than the original sentence distance in order to reduce data sparseness when conditioning on this feature.

[7] We do not show the number of anaphoric common NPs in the training+development data, as our model, being unsupervised, does not need to be trained on gold anaphoric common NPs.

[8] Condition (3) is introduced to ensure fairness in our comparison with Björkelund and Kuhn's resolver: we do not penalize its precision if it resolves a common NP to pronouns only.

Baseline	R	P	F
Closest antecedent	5.2	1.0	1.7
Closest subject	5.8	1.2	2.0
Closest NP with the same head	49.3	28.9	36.5
Closest subject with the same head	25.2	31.8	28.1
Björkelund and Kuhn (2014)	42.0	50.6	45.9

Table 2: Common NP resolution results of the baseline systems on the test set.

	Best Baseline			Our Model		
Source	R	P	F	R	P	F
Overall (3268)	42.0	50.6	45.9	46.2	47.5	46.8
NW (753)	41.0	49.8	45.0	43.0	57.5	49.2
MZ (834)	47.4	54.4	50.6	56.6	67.5	61.5
WB (200)	23.5	43.5	30.5	42.5	26.2	32.4
BN (982)	42.0	49.0	45.2	39.3	46.4	42.6
BC (370)	40.8	47.6	44.0	49.5	32.8	39.4
TC (129)	45.0	56.9	50.2	46.5	30.3	36.7

Table 3: Common NP resolution results of the best baseline and our unsupervised model on the test set.

that a resolved common NP n is correctly resolved if (1) it is anaphoric and (2) its closest non-pronominal antecedent according to the response chain also appears in the gold chain containing n. Note that their resolver is trained on the same CoNLL-2012 training and development sets that we used.

The results of the baseline systems are shown in Table 2. Several observations can be made about these results. First, among the rule-based resolvers, Baseline 3 achieves the best performance, outperforming Baselines 1, 2, and 4 by 34.8%, 34.5%, and 8.4% in F-score respectively. From their relative performance, we can conclude that as far as common NP resolution is concerned, (1) recency plays a greater role than salience; and (2) although head match is a strong indicator of coreference, it still suffers from relatively low precision. Also, comparing the rule-based resolvers and the supervised resolver, we can see that the best baseline is the supervised resolver, which outperforms the best rule-based baseline (Baseline 3) by 9.4% in F-score.

Our model. Table 3 shows the results of the best baseline (the supervised resolver) and our model on the entire test set (row 1) and each of the six sources (rows 2−7). The number within the parentheses after the source name indicates the number of anaphoric common NPs in each source. Despite the fact that our model is unsupervised, it yields a small, though statistically insignificant, improvement over the best baseline on the entire test set (0.9% in F-score), and significantly outperforms the best baseline on NW, MZ and WB.[9]

Ablation Experiments

Impact of $P(n_h|c_h, l{=}1)$, $P(c|k)$ and $P(l{=}1|k_c)$. Recall that our model is composed of these three probability terms. To investigate the contribution of each probability term to overall performance, we conduct ablation experiments. Specifically, in each ablation experiment, we remove exactly one probability term from the model and retrain it.

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

System	R	P	F
Full model	46.2	47.5	46.8
− $P(n_h|c_h, l{=}1)$	46.3	44.5	45.4
− $P(c|k)$	50.9	27.4	35.6
− $P(l{=}1|k_c)$	22.6	19.4	20.9

Table 4: Probability term ablation results.

System	R	P	F
Full model	46.2	47.5	46.8
− Feature 1	45.7	47.0	46.3
− Feature 2	45.6	46.9	46.2
− Feature 3	44.9	46.9	45.9
− Feature 4	46.1	47.4	46.7
− Feature 5	45.1	29.1	35.4

Table 5: Context feature ablation results.

Ablation results are shown in Table 4. As we can see, after ablating $P(n_h|c_h, l{=}1)$, F-score drops significantly by 1.4%. This result suggests that EM can be used to learn useful lexical relationships for common NP resolution from unannotated data. Furthermore, after ablating $P(c|k)$, F-score drops significantly by 11.2%. This justifies our use of $P(c|k)$, through which we distribute probability mass to all but the linguistically implausible candidate antecedents. Finally, after ablating $P(l{=}1|k_c)$, F-score decreases significantly by 25.9%. This result illustrates the importance of the context features in our model.

Context feature ablation. Recall that we employed five context features to encode the relationship between a common NP and a candidate antecedent. To determine the relative contribution of these five features to overall performance, we conduct ablation experiments. In these ablation experiments, both $P(n_h|c_h, l{=}1)$ and $P(c|k)$ are retained in the model.

Ablation results are shown in rows 2−6 of Table 5. To facilitate comparison, the F-score of the model in which all five context features are used is shown in row 1. As we can see, feature 5 (the feature encoding whether a candidate antecedent has the highest rank) is the most useful feature: its removal causes the F-score of our resolver to drop significantly by 11.4%. The remaining four features are also useful: ablating them causes F-score to drop by 0.1−0.9%, although only the drop caused by the removal of feature 3 (the feature encoding head match results) is significant.

Conclusion

We proposed an unsupervised probabilistic model for Chinese common NP resolution. To our knowledge, this is the first unsupervised model specifically designed for Chinese common NP resolution. Experiments on the OntoNotes 5.0 corpus showed that our unsupervised model rivaled its state-of-the-art supervised counterpart in performance.

Acknowledgments

We thank the three anonymous reviewers for their comments. 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.

References

Bergsma, S., and Lin, D. 2006. Bootstrapping path-based pronoun resolution. In *Proceedings of COLING/ACL*, 33--40.

Bergsma, S., and Yarowsky, D. 2011. NADA: A robust system for non-referential pronoun detection. In *Proceedings of DAARC*, 12--23.

Björkelund, A., and Kuhn, J. 2014. Learning structured perceptrons for coreference resolution with latent antecedents and non-local features. In *Proceedings of ACL (Volume 1: Long Papers)*, 47--57.

Charniak, E., and Elsner, M. 2009. EM works for pronoun anaphora resolution. In *Proceedings of EACL*, 148--156.

Chen, C., and Ng, V. 2012. Chinese noun phrase coreference resolution: Insights into the state of the art. In *Proceedings of COLING 2012: Posters Volume*, 185--194.

Chen, C., and Ng, V. 2013. Chinese zero pronoun resolution: Some recent advances. In *Proceedings of EMNLP*, 1360--1365.

Chen, C., and Ng, V. 2014a. Chinese overt pronoun resolution: A bilingual approach. In *Proceedings of AAAI*, 1615--1621.

Chen, C., and Ng, V. 2014b. Chinese zero pronoun resolution: An unsupervised approach combining ranking and integer linear programming. In *Proceedings of AAAI*, 1622--1628.

Chen, C., and Ng, V. 2014c. Chinese zero pronoun resolution: An unsupervised probabilistic model rivaling supervised resolvers. In *Proceedings of EMNLP*, 763--774.

Chen, C., and Ng, V. 2014d. SinoCoreferencer: An end-to-end Chinese event coreference resolver. In *Proceedings of LREC*, 4532--4538.

Cherry, C., and Bergsma, S. 2005. An expectation maximization approach to pronoun resolution. In *Proceedings of CoNLL*, 88--95.

Converse, S. 2006. *Pronominal Anaphora Resolution in Chinese*. Ph.D. Dissertation, University of Pennsylvania.

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

Ji, H., and Lin, D. 2009. Gender and animacy knolwedge discovery from web-scale n-grams for unsupervised mention detection. In *Proceedings of PACLIC*, 220--229.

Kehler, A.; Appelt, D. E.; Taylor, L.; and Simma, A. 2004. The (non) utility of predicate-argument frequencies for pronoun interpretation. In *HLT-NAACL 2004: Main Proceedings*, 289--296.

Kong, F., and Ng, H. T. 2013. Exploiting zero pronouns to improve Chinese coreference resolution. In *Proceedings of EMNLP*, 278--288.

Kong, F., and Zhou, G. 2010. A tree kernel-based unified framework for Chinese zero anaphora resolution. In *Proceedings of EMNLP*, 882--891.

Kong, F., and Zhou, G. 2012. Pronoun resolution in English and Chinese language based on tree kernel. *Journal of Software* 23(5):1085--1099.

Luo, X., and Zitouni, I. 2005. Multi-lingual coreference resolution with syntactic features. In *Proceedings of HLT/EMNLP*, 660--667.

Mitkov, R. 2002. *Anaphora Resolution*. Longman.

Pradhan, S.; Moschitti, A.; Xue, N.; Uryupina, O.; and Zhang, Y. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Proceedings of EMNLP-CoNLL: Shared Task*, 1--40.

Qian, X.; Zhang, Q.; Huang, X.; and Wu, L. 2010. 2d trie for fast parsing. In *Proceedings of COLING*, 904--912.

Rahman, A., and Ng, V. 2009. Supervised models for coreference resolution. In *Proceedings of EMNLP*, 968--977.

Rahman, A., and Ng, V. 2011. Narrowing the modeling gap: A cluster-ranking approach to coreference resolution. *Journal of Artificial Intelligence Research* 40:469--521.

Robert, P.; David, G.; Ke, C.; Junbo, K.; and Kazuaki, M. 2009. Chinese Gigaword fourth edition. Linguistic Data Consortium, Philadelphia.

Song, W.; Qin, B.; Lang, J.; and Liu, T. 2008. Combining syntax and word sense for Chinese pronoun resolution. *Journal of Chinese Information Processing* 22(6):8--13.

Stoyanov, V.; Gilbert, N.; Cardie, C.; and Riloff, E. 2009. Conundrums in noun phrase coreference resolution: Making sense of the state-of-the-art. In *Proceedings of ACL-IJCNLP*, 656--664.

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

Wang, C.-S., and Ngai, G. 2006. A clustering approach for unsupervised Chinese coreference resolution. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, 40--47.

Xue, N.; Xia, F.; Chiou, F.-D.; and Palmer, M. 2005. The Penn Chinese Treebank: Phrase structure annotation of a large corpus. *Natural Language Engineering* 11(02):207--238.

Yang, X.; Su, J.; and Tan, C. L. 2005. Improving pronoun resolution using statistics-based semantic compatibility information. In *Proceedings of ACL*, 165--172.

Yeh, C.-L., and Chen, Y.-C. 2007. Zero anaphora resolution in Chinese with shallow parsing. *Journal of Chinese Language and Computing* 17(1):41--56.

Zhao, S., and Ng, H. T. 2007. Identification and resolution of Chinese zero pronouns: A machine learning approach. In *Proceedings of EMNLP-CoNLL*, 541--550.