Chinese Event Coreference 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

Recent work has successfully leveraged the semantic information extracted from lexical knowledge bases such as WordNet and FrameNet to improve English event coreference resolvers. The lack of comparable resources in other languages, however, has made the design of high-performance non-English event coreference resolvers, particularly those employing unsupervised models, very difficult. We propose a generative model for the under-studied task of Chinese event coreference resolution that rivals its supervised counterparts in performance when evaluated on the ACE 2005 corpus.

1 Introduction

Event coreference resolution is the task of determining which event mentions in a text refer to the same real-world event. Compared to entity coreference, event coreference is not only much less studied, but it is arguably more challenging. Recall that for two event mentions to be coreferent, both their *triggers* (i.e., the words realizing the occurrence of the events) and their corresponding *arguments* (e.g., the times, places, and people involved in them) have to be compatible. However, identifying potential arguments (which is typically performed by an entity extraction system), linking arguments to their event mentions (which is typically performed by an event extraction system), and determining the compatibility between two event arguments (which is the job of an entity coreference resolver), are all non-trivial tasks. In other words, end-to-end event coreference resolution is complicated in part by the fact that an event coreference resolver has to rely on the noisy outputs produced by its upstream components in the standard information extraction (IE) pipeline.

In this paper, we examine Chinese event coreference resolution. While English event coreference is under-investigated, Chinese event coreference is much less studied than English event coreference. In terms of task definition, there is no difference between English and Chinese event coreference. However, the design of high-performance Chinese event coreference resolvers is complicated in part by the lack of large-scale lexical knowledge bases. Recent work by Bejan and Harabagiu (2010; 2014) has shown that the semantic information extracted from WordNet (Fellbaum, 1998) and FrameNet (Baker et al., 1998) significantly contributed to the performance of their English event coreference resolver.

While the lack of comparable lexical knowledge bases in Chinese can be mitigated in part by the use of event coreference annotated data, we focus on a challenging version of the task --- *unsupervised* Chinese event coreference resolution. Specifically, our goal is to learn an event coreference model *without* using data annotated with event coreference links. When evaluated on the Chinese portion of the ACE 2005 corpus, our unsupervised probabilistic model for event coreference resolution rivals its state-of-the-art supervised counterpart in performance. This, together with the fact that its underlying generative process is not language-dependent and does not rely on features extracted from lexical knowledge bases, potentially enables it to be applied to languages where neither annotated data nor large-scale

Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL, pages 1097–1107,
Denver, Colorado, May 31 – June 5, 2015.

knowledge bases are available.

Another feature of our model that deserves mention is that it performs joint event coreference resolution and anaphoricity determination. Anaphoricity determination, the task of determining whether a mention is anaphoric and hence needs to be resolved, is an issue common to both entity and event coreference resolution. However, determining the anaphoricity of an event mention is arguably more difficult than determining the anaphoricity of a pronoun. 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 event mentions makes the identification of anaphoric event mentions 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 event mention is not a viable option, as only 24.4% of the Chinese event mentions in our evaluation corpus (ACE 2005) are anaphoric. Our decision to jointly model anaphoricity determination and event coreference resolution was inspired by the difficulty of designing a standalone system for determining the anaphoricity of event mentions.

2 Related Work

Almost all existing approaches to event coreference are developed for English. These approaches can broadly be divided into three categories.

Within-document coreference is the most popularly investigated and arguably the most important event coreference task. While early work in MUC (e.g., Humphreys et al. (1997)) is limited to several scenarios, ACE takes a further step towards processing more fine-grained events. Most ACE event coreference resolvers are supervised, training a pairwise model to determine whether two event mentions are coreferent (e.g., Ahn (2006)).

Improvements to this standard approach include the use of *feature weighting* to train a better model (McConky et al., 2012), and *graph-based clustering algorithms* to produce event coreference clusters (Chen and Ji, 2009; Sangeetha and Arock, 2012). Chen et al. (2011) train multiple classifiers to handle coreference between event mentions of different syntactic types (e.g., verb-noun coreference, noun-noun coreference) on the OntoNotes corpus (Pradhan et al., 2007). However, OntoNotes is only partially annotated with event coreference links, and Chen et al. further make the simplifying assumption that event coreference chains are all and only those coreference chains that involve at least one verb.

More recently, Cybulska and Vossen (2012) and Goyal et al. (2013) have performed event coreference using semantic relations (e.g., hyponymy relations extracted from WordNet) and distributional semantic information, respectively, on the Intelligence Community (IC) corpus (Hovy et al., 2013). The IC corpus, which at the time of writing is not yet publicly available, is different from the MUC and ACE corpora in that it is annotated with not only *full* event coreference relations but also *partial* event coreference relations. Partial coreference is a term coined by Hovy et al. to refer to event relations that exhibit subtle deviation from the perfect identity of events (e.g., the subset relation, the membership relation). While all of the aforementioned work addresses the full event coreference task, a two-stage approach is recently proposed by Araki et al. (2014) to identify subevent relations from the IC corpus.

Cross-document coreference is first investigated by Bagga and Baldwin (1999), who represent an event mention as a vector of its context words and determine whether two event mentions are coreferent based on the cosine similarity of their vectors. Bejan and Harabagiu (2010; 2014) and Lee et al. (2012) propose nonparametric models and a joint entity and event coreference model respectively for within- and cross-document event coreference, evaluating their models on the ECB corpus. However, ECB "is annotated mainly for cross-document coreference" and many difficult cases of within-document coreference are not annotated (Liu et al., 2014).

Naughton (2009) and Elkhlifi and Faiz (2009) have worked on **sentence-level event coreference**, where the goal is to determine whether two sentences containing event mentions are coreferent. Somewhat unfortunately, simplifying assumptions have to be made when a sentence containing multiple non-coreferent event mentions is encountered.

Compared to English event coreference, there has been much less work on Chinese event coreference. SinoCoreferencer (Chen and Ng, 2014), a publicly-available ACE-style within-document event corefer-

ence resolver for Chinese that achieves state-of-the-art results, employs a supervised approach where a classifier is trained to determine whether two event mentions are coreferent. We will compare our unsupervised model against this supervised resolver.

3 ACE Event Coreference

In this section, we overview the ACE 2005 event coreference task, which is the version of the within-document event coreference task we focus on.

The ACE 2005 event coreference task requires that an event coreference resolver perform coreference on event mentions belonging to one of the ACE event types. More specifically, an event mention is composed of a *trigger* (i.e., the word realizing the event's occurrence) and a set of *arguments* (i.e., the event's participants). Each event trigger has a *type* and a *subtype*. In ACE 2005, eight event types are defined, which are further subcategorized into 33 subtypes. Each event argument has a semantic *role*. In ACE 2005, a set of argument roles is defined for each event type. That is, an event's type determines what roles its mentions' arguments can assume. Not surprisingly, two event mentions cannot be coreferent if their triggers have different subtypes or they have incompatible arguments (e.g., their dates or locations are different).

To better understand the ACE 2005 event coreference task, consider the sentence in Figure 1, which is taken from the ACE 2005 corpus. This example contains three event mentions belonging to the ACE event types. Specifically, these three mentions are triggered by the words 离 (leaving), 暗杀 (assassinated) and 攻击 (attack). 暗杀 and 攻击 have type LIFE and subtype DIE, whereas 离 has type MOVEMENT and subtype TRANSPORT. Note that 暗杀 and 攻击 refer to the same real-world event and are therefore coreferent.

沙米里与其子在上午交通尖峰时间 [离] 家时，遭到 [暗杀]。这次 [攻击] 再次显示叛乱分子能力。

Shameri and his son were [assassinated] during morning rush hour when [leaving] home. This [attack] once again demonstrated the insurgents' ability.

Figure 1: An excerpt from a Chinese document in the ACE 2005 corpus with the corresponding English translation. The event mentions are bracketed.

4 The Generative Model

In this section, we present our generative model.

4.1 Notation

We begin by introducing the notation that we use in the rest of this paper. We denote e to be the current event mention to be resolved (henceforth the *active* event mention). C, the set of candidate antecedents of e, contains all the event mentions preceding e in the associated text as well as a dummy candidate antecedent d (to which e will be resolved if it is non-anaphoric). Also, we define k to be the context surrounding e as well as every candidate antecedent c in C, and k_c to be the context surrounding e and candidate antecedent c. Moreover, we define l to be a binary variable indicating whether c is the correct antecedent of e. Finally, e_t and c_t denote e and c's respective trigger words.

4.2 Training

Our model estimates $P(e, k, c, l)$, the probability of seeing (1) the active event mention e; (2) the context k surrounding e and its candidate antecedents; (3) a candidate antecedent c of e; and (4) l, a binary value indicating whether c is e's correct antecedent. Since we estimate this probability from a raw, unannotated corpus, we are effectively treating e, 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 Expectation-Maximization (EM) algorithm (Dempster et al., 1977). Specifically, we use EM to iteratively estimate the model parameters from data in which each event mention is labeled with the probability that it corefers with each of its candidate antecedents, and apply the resulting model to relabel each event mention 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.

4.2.1 E-Step

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

its correct antecedent, we can rewrite $P(l{=}1|e,k,c)$ as follows:

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

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

$$P(e,k,c,l{=}1) = P(e|k,c,l{=}1) * P(l{=}1|k,c) * P(c|k) * P(k) \quad (2)$$

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

$$P(e|k,c,l{=}1) \approx P(e_t|c_t,l{=}1) \quad (3)$$

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

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

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

$$\frac{P(k_c|l{=}1)P(l{=}1)}{P(k_c|l{=}1)P(l{=}1) + P(k_c|l{=}0)P(l{=}0)} \quad (5)$$

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 (5) as:

$$\frac{\prod_i P(f_c^i|l{=}1)P(l{=}1)}{\prod_i P(f_c^i|l{=}1)P(l{=}1) + \prod_i P(f_c^i|l{=}0)P(l{=}0)} \quad (6)$$

Given Equations (2), (3), (4) and (6), we can rewrite $P(l{=}1|e,k,c)$ as follows:

$$P(l{=}1|e,k,c) = \frac{P(e,k,c,l{=}1)}{\sum_{c' \in C} P(e,k,c',l{=}1)} \approx \frac{P(e_t|c_t,l{=}1) * \frac{\prod_i P(f_c^i|l{=}1)}{Z_c} * P(c|k)}{\sum_{c' \in C} P(e_t|c'_t,l{=}1) * \frac{\prod_i P(f_{c'}^i|l{=}1)}{Z_{c'}} * P(c'|k)} \quad (7)$$

where

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

As we can see from Equation (7), our model has four groups of parameters, namely $P(e_t|c_t,l{=}1)$, $P(f_c^i{=}1|l)$, $P(l)$ and $P(c|k)$. With these four groups of parameters, we can apply Equation (7) to efficiently compute $P(l{=}1|e,k,c)$.

Two points deserve mention before we describe our M-step. First, among the four groups of parameters, $P(e_t|c_t,l{=}1)$ and $P(f_c^i|l)$ are estimated in the M-step described below; $P(l)$ is estimated in parameter initialization and used throughout the EM iterations (details on parameter initialization appear after the M-step); and $P(c|k)$ is computed heuristically. 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. To identify such candidate antecedents, we employ a simple heuristic, which considers a candidate antecedent implausible if its event subtype is different from that of e. Consequently, we model $P(c|k)$ as follows: if c is implausible, we set $P(c|k)$ to 0 and distribute the probability mass uniformly over all and only the plausible candidate antecedents. Since this heuristic is not applicable to dummy candidates, we assume for simplicity that they are all plausible.

Second, by including d as a dummy candidate antecedent for each e, we model anaphoricity determination and event coreference in a joint fashion. If the model resolves e to d, it means that the model posits e as non-anaphoric; on the other hand, if the model resolves e to a non-dummy candidate antecedent c, it means that the model posits e as anaphoric and c as e's correct antecedent. This joint modeling method has proven effective in earlier work on supervised entity coreference resolution (e.g., Rahman and Ng (2009; 2011)).

4.2.2 M-Step

Given $P(l{=}1|e,k,c)$, the goal of the M-step is to (re)estimate two of the four groups of parameters mentioned above, namely $P(e_t|c_t,l{=}1)$ and $P(f_c^i|l)$, using maximum likelihood estimation.

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

$$P(e_t|c_t, l{=}1) = \frac{Count(e_t, c_t, l{=}1) + \theta}{Count(c_t, l{=}1) + \theta * |t|} \quad (9)$$

where $Count(c_t, l{=}1)$ is the expected number of times c has trigger word c_t when it is the antecedent of an event mention; and $|t|$ is the number of possible trigger words in the training data (we treat the "trigger word" of a dummy candidate antecedent as an unseen word). Also, θ is the Laplace smoothing parameter, which we set to 1, and $Count(e_t, c_t, l{=}1)$ is the expected number of times e has e_t as its trigger when its antecedent c has trigger c_t. Given trigger words e'_t and c'_t, we compute $Count(e'_t, c'_t, l{=}1)$ as follows:

$$Count(e'_t, c'_t, l{=}1) = \sum_{e,c:e_t=e'_t,c_t=c'_t} P(l{=}1|e, k, c) \quad (10)$$

The remaining group of parameters, $P(f_c^i|l)$, can be estimated in a similar fashion.

To start the induction process, we initialize all parameters with uniform values. Specifically, $P(e_t|c_t, l{=}1)$ is set to $\frac{1}{|t|}$, and $P(l{=}1|k_c)$ is set to 0.5. As noted before, $P(l)$ is also initialized here and used throughout the EM iterations. Recall that $P(l{=}1)$ is the fraction of event pairs that are coreferent. Since we assumed earlier that each event mention has exactly one (dummy or non-dummy) antecedent, $P(l{=}1)$ can be computed as the number of event mentions divided by the total number of event pairs. After initialization, we iteratively run the E-step and the M-step until convergence.

There is an important question we have not addressed: what features f_c^i should we use to represent context k_c, which we need to estimate $P(f_c^i|l)$? We defer the discussion of this question to Section 5.

4.3 Inference

After training, we can apply the resulting model to resolve event mentions. Given an event mention e, we determine its antecedent as follows:

$$\hat{c} = \arg\max_{c \in C} P(l{=}1|e, k, c) \quad (11)$$

where C is the set of candidate antecedents of e. In other words, we apply Equation (11) to each of e's candidate antecedents, and select the one that yields the largest probability. If c is a non-dummy candidate antecedent, we posit c as the antecedent of e; otherwise, we posit e as non-anaphoric.

5 Context Features

As mentioned at the end of Section 4.2.2, to fully specify our model, we need to describe the features f_c^i used to represent k_c, which is needed to compute $P(f_c^i|l)$. Recall that k_c encodes the context surrounding candidate antecedent c and active event mention e. We represent k_c using six features that encode the relationship between c and e, some of which are motivated by previous work on supervised event coreference resolution (e.g., Chen and Ji (2009)). Below we describe these six features, which can be broadly divided into three categories.

5.1 Trigger-Based Features

We employ two trigger-based features (Features 1 and 2), both of which are binary-valued and are computed based on e's and c's triggers.

Feature 1 encodes whether c_t and e_t, the trigger words of c and e, satisfy any of the following three conditions:

1. c_t and e_t are lexically identical;
2. c_t and e_t contain the same basic verb (BV) and their verb structures are compatible;
3. the similarity between c_t and e_t is greater than a certain threshold (which we set to 0.8 in our experiments).

Intuitively, Feature 1 is a recall-enhancing feature: it encodes a condition whose satisfaction can help discover many event coreference links. However, it is not designed to be precision-oriented, as it is computed based solely on the triggers and not their surrounding contexts. Below we explain conditions 2 and 3 in more detail.

Recall that condition 2 encodes our observation that an event coreference relation may exist between two non-identical trigger words having the same BV if their verb structures are compatible. To understand this condition, let us explain the notion of BVs and how we determine the compatibility of two verb structures. A BV is a single-character Chinese verb, which is the building block of all Chinese verbs.

Specifically, Li et al. (2012) observe that, with a few exceptions, a Chinese verb constructed out of a basic verb bv possesses one of six main *verb structures*: (1) bv (e.g., 逮 (arrest)); (2) bv + verb (e.g., 送到 (deliver), where bv is 送); (3) verb + bv (e.g., 离开 (leave), where bv is 开); (4) bv + complementation (e.g., 进了 (enter), where bv is 进); (5) bv + noun/adjective (e.g., 开枪 (shoot), where bv is 开); (6) noun/adjective + bv (e.g., 轻伤 (slight wound), where bv is 伤). Now, assuming that t_1 and t_2 are two lexically different trigger words containing the same BV (bv), we say that their verb structures (denoted as vs_1 and vs_2) are incompatible if one of the following conditions is satisfied: (1) bv appears in different positions in t_1 and t_2 (e.g., 开枪 (shoot) and 离开 (leave), where bv is 开); (2) both vs_1 and vs_2 have bv + verb or verb + bv as their verb structure (e.g., 送到 (deliver) and 赶到 (reach), where bv is 到); or (3) both vs_1 and vs_2 have noun/adjective + bv or bv + noun/adjective as their verb structure (e.g., 轻伤 (slight wound) and 重伤 (severe wound), where bv is 伤). Note that these three incompatibility conditions encode our commonsense knowledge of when two Chinese verbs having the same BV cannot have the same meaning.

Next, we explain how we compute the similarity between two trigger words in condition 3. To capture their semantic similarity, we first apply word2vec (Mikolov et al., 2013) to the Chinese Gigaword corpus (Parker et al., 2009) to obtain a vector representation of each word and then compute the cosine similarity between the two word vectors.

Feature 2, our second trigger-based feature, encodes whether two nominal event mentions are incompatible w.r.t. number. Specifically, its value is True if and only if (1) c and e are both nouns, and (2) one is singular and the other is plural. Intuitively, this feature encodes a non-coreference condition.

5.2 Argument-Based Features

We employ three argument-based features (Features 3−5), all of which are binary-valued and are computed based on c's and e's arguments.

Feature 3 encodes whether c and e possess two arguments that have the same semantic role but different semantic classes.[1] Intuitively, Feature 3 encodes a non-coreference condition: c and e cannot be coreferent if such arguments exist.

Feature 4 can be viewed as a generalized version of Feature 3, encoding whether c and e possess two arguments that have the same semantic role but are not coreferent.

Feature 5 encodes whether c and e possess two named entity (NE) arguments that both have VALUE as their NE type but are lexically different. Such event mentions have a good chance of being not coreferent.

5.3 Distance Feature

We employ one distance feature (Feature 6) that encodes how far c and e are apart from each other in terms of the number of event mentions. To reduce data sparseness during parameter estimation, however, we quantize the distance as follows. Let d be the distance between the first event mention and the last event mention in the document for which the distance feature will be computed. Note that the distance between an arbitrary pair of event mentions in this document will be between 0 and d. We divide the interval [0,d] into four equal-sized regions, and set the value of the distance feature based on which of the four bins it falls into.

5.4 Features for Dummy Candidates

Now that we can compute the aforementioned six features for a non-dummy candidate antecedent, we next specify how we compute these features for a dummy candidate antecedent d of active event mention e. For Feature 1, we set the feature value of d to True, whereas for Features 2−5, we set the feature value of d to False. To understand why these values are chosen, note that for each of these features the opposite value could be a strong indicator of non-coreference, potentially causing the model to have an overly strong bias against selecting d as the antecedent of e.

Finally, to compute Feature 6, we assume that d is the zero-th event mention of the associated document, and then compute the distance feature in the same way as described above. By letting d be the zero-th event mention, we make the probability of picking d as the correct antecedent (the probability of

[1] The possible semantic classes are the ACE 2005 entity types, i.e., PERSON, ORGANIZATION, GPE, FACILITY, and LOCATION.

classifying e as non-anaphoric) depend on e's position in the associated text. This makes sense because in general, the probability of e being non-anaphoric tends to be larger (smaller) when it appears earlier (later) in the document.

6 Evaluation

6.1 Experimental Setup

Dataset. For evaluation, we conduct five-fold cross-validation experiments on the 633 Chinese documents of the ACE 2005 training corpus. Statistics on the corpus are shown in Table 1.

Evaluation measures. We report results in terms of recall (R), precision (P), and F-score (F) using the commonly-used coreference evaluation measures given by the CoNLL scorer, namely the link-based MUC scorer (Vilain et al., 1995), the mention-based B^3 scorer (Bagga and Baldwin, 1998), the entity-based version of the CEAF scorer (Luo, 2005), and the Rand index-based BLANC scorer (Recasens and Hovy, 2011), after singleton event mentions are removed from the coreference partitions produced by our resolver. We use the latest version (version 8) of the CoNLL scorer[2], which fixes a bug in previous versions (Pradhan et al., 2014). In addition, we report the CoNLL score (Pradhan et al., 2011), which is the unweighted average of the MUC, B^3, and CEAF F-scores.

Evaluation setting. We perform an end-to-end evaluation, as it can more accurately reflect the performance of an event coreference resolver when it is used in practice.

More specifically, to extract the event mentions used in our evaluation, we employ SinoCoreferencer[3], which, as mentioned before, is an end-to-end ACE-style Chinese IE system that achieves state-of-the-art event coreference results. Specifically, the event triggers needed to compute the trigger-based context features are extracted using SinoCoreferencer's event extraction subsystem. The event subtypes needed to identify and filter out implausible candidate antecedents are also provided by its event extraction subsystem. The event arguments needed to compute the argument-based context features are first extracted and typed by its entity extraction subsystem, and then linked to their triggers by its event extraction subsystem. Finally, the entity coreference links and the semantic roles needed to compute Feature 4 are provided by its entity coreference subsystem and its event extraction subsystem, respectively.[4] Details of each of these subsystems can be found in Chen and Ng (2014).

Documents	633
Sentences	9,967
Event mentions	3,333
Event coreference chains	2,521

Table 1: Statistics on the ACE 2005 Chinese corpus.

6.2 Results

We employ two supervised resolvers as baseline systems. The first baseline employs rote learning, simply positing two event mentions as coreferent if their corresponding triggers are annotated as coreferent in the training data. The second baseline is SinoCoreferencer, which has produced the best Chinese event coreference results to date on the ACE corpus.

Row 1 of Table 2 shows the results of the baseline that employs rote learning. As we can see, this baseline achieves a CoNLL score of 37.9. Row 2 shows the results of SinoCoreferencer. It performs significantly better than the rote-learning baseline w.r.t. all five scoring measures[5], achieving a CoNLL score of 39.2. Finally, row 3 shows the results of our model. Despite being unsupervised, it significantly outperforms the better baseline, SinoCoreferencer, w.r.t. all five scoring measures, achieving a CoNLL score of 41.5, which is 2.3 points higher than that of SinoCoreferencer. These results suggest that a generative approach to unsupervised event coreference holds promise.

6.3 Ablation Experiments

Recall that in our model eight probability terms play a major role: $P(e_t|c_t)$, $P(c|k)$, and $P(f_c^i|l)$ for each

[2] conll.github.io/reference-coreference-scorers/

[3] Downloadable from http://www.hlt.utdallas.edu/~yzcchen/coreference/

[4] We employ only those semantic roles that can be reliably determined by SinoCoreferencer's event extraction subsystem, namely, AGENT, ADJUDICATOR, DEFENDANT, GIVER, PERSON, PLACE, POSITION, ORGANIZATION, ORIGIN, and RECIPIENT.

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

System	MUC			B^3			$CEAF_e$			BLANC			CoNLL
	R	P	F	R	P	F	R	P	F	R	P	F	F
Rote learning	42.6	36.4	39.3	41.4	32.3	36.3	37.0	39.7	38.3	27.4	20.0	23.1	37.9
SinoCoreferencer	42.7	38.3	40.4	41.5	34.7	37.8	39.9	39.2	39.5	28.1	23.7	25.7	39.2
Our model	43.1	42.4	42.8	41.4	39.1	40.2	40.7	42.6	41.6	27.5	26.4	26.9	41.5

Table 2: Five-fold cross-validation event coreference results on the ACE 2005 corpus.

of the six context features. To investigate the contribution of each probability term to overall performance, we conduct ablation experiments. Specifically, in each ablation experiment, we remove exactly one term from the model and retrain it.

Ablation results are shown in Table 3. Each row contains the F-scores obtained via the five evaluation measures. To facilitate comparison, the scores of the model in which all eight probability terms are used is shown in row 1. As we can see, Feature 1 is the most useful feature: its removal causes the CoNLL score to drop significantly by 5.3 points. A closer examination reveals that the drop in the CoNLL score is caused by a significant drop in recall w.r.t. all scorers. Recall that this feature encodes the conditions under which two triggers are likely to be coreferent. It is perhaps not surprising that its removal causes a significant drop in recall.

The second most useful feature is $P(c|k)$, which places zero probability mass on candidate antecedents whose event subtypes are different from that of the active event mention. Its removal causes the CoNLL score to drop significantly by 1.6 points. The removal of each other feature resulted in a small, insignificant drop in the CoNLL score.

System	MUC	B^3	$CEAF_e$	BLANC	CoNLL
Full model	42.8	40.2	41.6	26.9	41.5
$-\ P(e_t\|c_t)$	42.9	39.8	40.9	26.9	41.2
$-\ P(c\|k)$	41.2	38.6	39.8	24.9	39.9
– Feature 1	37.5	32.9	38.2	20.8	36.2
– Feature 2	42.5	39.9	41.4	26.6	41.3
– Feature 3	42.4	40.0	41.3	26.9	41.2
– Feature 4	42.5	40.1	41.7	27.0	41.4
– Feature 5	42.4	40.0	41.4	26.5	41.3
– Feature 6	42.3	39.6	40.9	26.8	40.9

Table 3: Ablation results in terms of F-scores.

6.4 Error Analysis

It is somewhat surprising that our unsupervised event coreference model outperforms the better supervised baseline, SinoCoreferencer. To understand why, we analyze the errors made by the two resolvers.

Our analysis proceeds as follows. First, to gain insights into the differences between the two resolvers, we examine those candidate event mentions that are correctly handled by one model but not the other (Section 6.4.1). Specifically, we consider a candidate event mention e correctly handled if (1) e is a correctly resolved anaphoric event mention; (2) e is an unresolved singleton event mention; or (3) e is an unresolved non-event mention (i.e., not a true event mention). Second, to understand how to improve event coreference, we identify the major sources of error made by both resolvers (Section 6.4.2).

6.4.1 Common Sources of Disagreement

There are 323 candidate event mentions that are correctly handled by one model but not the other in our dataset. Among these 323 cases, 205 (50 anaphoric + 79 singletons + 76 non-event) are correctly handled by the unsupervised model, and 118 (42 anaphoric + 52 singletons + 24 non-event) are correctly handled by SinoCoreferencer.

From these numbers, we can see that the unsupervised model performs far better than SinoCoreferencer in *not* resolving the singletons and the non-event mentions. This is perhaps not surprising given the unsupervised model's relatively stricter conditions on resolving a candidate event mention. Specifically, it is unlikely to posit two candidate event mentions as coreferent unless (1) their triggers have a BV match or a large word2vec similarity value and (2) none of the non-coreference conditions are satisfied. Overall, these results explain why the unsupervised model has a much higher precision than SinoCoreferencer.

Not only does the unsupervised model perform much better in not resolving singletons and non-event mentions, but it is also slightly more accurate

in resolving the anaphoric event mentions, which ultimately enables it to achieve a higher recall than SinoCoreferencer. In particular, it correctly resolves 50 anaphoric mentions that are incorrectly handled by SinoCoreferencer. The successful resolution of these anaphoric mentions can be attributed largely to its use of BV and word2vec, neither of which is exploited by SinoCoreferencer. However, while a BV match or a high word2vec similarity value is a good indicator of event coreference, they are by no means perfect. This partly explains why there are singletons and non-event mentions that are correctly handled by SinoCoreferencer but not the unsupervised model.

Despite the fact that SinoCoreferencer slightly lags behind the unsupervised model in resolving anaphoric mentions, it correctly resolves 42 anaphoric event mentions that are incorrectly handled by the unsupervised model. These are cases that cannot be handled simply by relying on BV match or word2vec similarity. More specifically, two of the unique features of SinoCoreferencer are primarily responsible for its successful resolution of these event mentions. First, it learns coreferent trigger pairs from the training data. These pairs proved to be useful for event coreference, as we saw from the competitive results provided by the rote-learning baseline. Second, unlike the unsupervised model, SinoCoreferencer can posit two event mentions as coreferent *without* considering their triggers. More specifically, SinoCoreferencer may posit two event mentions as coreferent if the corresponding arguments of the two event mentions (i.e., arguments having the same role) are coreferent. Neither of these two recall-enhancing features of SinoCoreferencer is a precise indicator of event coreference. In other words, employing them widens the precision gap between the two resolvers.

6.4.2 Common Sources of Error

Next, we discuss the major sources of error made by both our unsupervised model and SinoCoreferencer. Broadly, the errors can be divided into two categories, precision errors and recall errors.

Precision errors arise primarily from erroneous coreference links established between (1) one or more candidate event mentions that are not true event mentions; (2) two event mentions with incompatible latent attributes such as MODALITY, POLARITY, GENERICITY, and TENSE, since these attributes are not exploited by the two resolvers; (3) two event mentions with incompatible arguments, since these arguments fail to be extracted by the argument identification component; (4) two mentions representing events that occur at different times, since the event mentions are not timestamped[6]; and (5) two event mentions whose corresponding arguments are incorrectly posited by the entity coreference subsystem as coreferent.

On the other hand, recall errors arise primarily from missing coreference links attributed to (1) the trigger identification component's failure to detect one or both of the triggers involved in an event coreference link; (2) the entity coreference subsystem's failure to establish the link(s) between the corresponding arguments of two coreferent event mentions; (3) the lack of positive evidence of event coreference, such as BV match, high word2vec similarity, and coreferent arguments; and (4) the argument identification component's failure to extract one or more arguments of an event mention.

7 Conclusions

We presented a generative model for the relatively under-studied task of unsupervised Chinese event coreference resolution whose parameters were learned using EM from an unannotated corpus. When evaluated on the ACE 2005 corpus, our model significantly outperforms SinoCoreferencer, a state-of-the-art Chinese event coreference resolver.

Since the performance of our resolver is limited in part by the errors made by SinoCoreferencer's subsystems, we plan to mitigate this problem by performing joint inference for entity coreference, event extraction and event coreference in future work.

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.

[6] Not all of these errors can be fixed by exploiting the TENSE attribute, as TENSE is only a crude approximation of time. For instance, in the phrases 首度被捕 (arrested for the first time) and 再度被捕 (arrested again), the two occurrences of 被捕 (arrested) are associated with different timestamps despite the fact that they have the same TENSE.

References

David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pages 1--8.

Jun Araki, Zhengzhong Liu, Eduard Hovy, and Teruko Mitamura. 2014. Detecting subevent structure for event coreference resolution. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, pages 4553--4558.

Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *Proceedings of the Linguistic Coreference Workshop at the First International Conference on Language Resources and Evaluation*, page 563--566.

Amit Bagga and Breck Baldwin. 1999. Cross-document event coreference: Annotation, experiments, and observations. In *Proceedings of the ACL Workshop on Coreference and Its Applications*, pages 1--9.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics, Volume 1*, pages 86--90.

Cosmin Bejan and Sanda Harabagiu. 2010. Unsupervised event coreference resolution with rich linguistic features. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1412--1422.

Cosmin Bejan and Sanda Harabagiu. 2014. Unsupervised event coreference resolution. *Computational Linguistics*, 40(2):311--347.

Shane Bergsma and David Yarowsky. 2011. NADA: A robust system for non-referential pronoun detection. In *Proceedings of the 8th Discourse Anaphora and Anaphor Resolution Colloquium*, pages 12--23.

Zheng Chen and Heng Ji. 2009. Graph-based event coreference resolution. In *Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing (TextGraphs-4)*, pages 54--57.

Chen Chen and Vincent Ng. 2014. SinoCoreferencer: An end-to-end Chinese event coreference resolver. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, pages 4532--4538.

Bin Chen, Jian Su, Sinno Jialin Pan, and Chew Lim Tan. 2011. A unified event coreference resolution by integrating multiple resolvers. In *Proceedings of the 5th International Joint Conference on Natura Language Processing*, pages 102--110.

Agata Cybulska and Piek Vossen. 2012. Using semantic relations to solve event coreference in text. In *Proceedings of the LREC Workshop on Semantic Relations-II Enhancing Resources and Applications (SemRel 2012)*, pages 60--67.

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.

Aymen Elkhlifi and Rim Faiz. 2009. Automatic annotation approach of events in news articles. *International Journal of Computing and Information Sciences*, 7(1):40--50.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.

Kartik Goyal, Sujay Kumar Jauhar, Huiying Li, Mrinmaya Sachan, Shashank Srivastava, and Eduard Hovy. 2013. A structured distributional semantic model for event co-reference. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: Volume 2 (Short Papers)*, pages 467--473.

Eduard Hovy, Teruko Mitamura, Felisa Verdejo, Jun Araki, and Andrew Philpot. 2013. Events are not simple: Identity, non-identity, and quasi-identity. In *Proceedings of the NAACL-HLT Workshop on Events: Definition, Detection, Coreference, and Representation*, pages 21--28.

Kevin Humphreys, Robert Gaizauskas, and Saliha Azzam. 1997. Event coreference for information extraction. In *Proceedings of the ACL/EACL Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*, pages 75--81.

Heeyoung Lee, Marta Recasens, Angel Chang, Mihai Surdeanu, and Dan Jurafsky. 2012. Joint entity and event coreference resolution across documents. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 489--500.

Peifeng Li, Guodong Zhou, Qiaoming Zhu, and Libin Hou. 2012. Employing compositional semantics and discourse consistency in Chinese event extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1006--1016.

Zhengzhong Liu, Jun Araki, Eduard Hovy, and Teruko Mitamura. 2014. Supervised within-document event coreference using information propagation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, pages 4539--4544.

Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 25--32.

Katie McConky, Rakesh Nagi, Moises Sudit, and William Hughes. 2012. Improving event co-reference by context extraction and dynamic feature weighting. In *Proceedings of the 2012 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support*, pages 38--43.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 3111--3119.

Martina Naughton. 2009. *Sentence Level Event Detection and Coreference Resolution*. Ph.D. thesis, National University of Ireland, Dublin, Ireland.

Robert Parker, David Graff, Ke Chen, Junbo Kong, and Kazuaki Maeda. 2009. Chinese Gigaword fourth edition. Linguistic Data Consortium, Philadelphia, PA.

Sameer Pradhan, Lance Ramshaw, Ralph Weischedel, Jessica MacBride, and Linnea Micciulla. 2007. Unrestricted coreference: Identifying entities and events in OntoNotes. In *Proceedings of the International Conference on Semantic Computing*, pages 446--453.

Sameer Pradhan, Lance Ramshaw, Mitchell Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. 2011. CoNLL-2011 Shared Task: Modeling unrestricted coreference in OntoNotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1--27.

Sameer Pradhan, Xiaoqiang Luo, Marta Recasens, Eduard Hovy, Vincent Ng, and Michael Strube. 2014. Scoring coreference partitions of predicted mentions: A reference implementation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 30--35.

Altaf Rahman and Vincent Ng. 2009. Supervised models for coreference resolution. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 968--977.

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

Marta Recasens and Eduard Hovy. 2011. BLANC: Implementing the Rand Index for coreference evaluation. *Natural Language Engineering*, 17(4):485--510.

S. Sangeetha and Michael Arock. 2012. Event coreference resolution using mincut based graph clustering. In *Proceedings of the Fourth International Workshop on Computer Networks & Communications*, pages 253--260.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Proceedings of the Sixth Message Understanding Conference*, pages 45--52.