Chinese Event Coreference Resolution: An Unsupervised Probabilistic Model Rivaling Supervised Resolvers

Chen Chen, Vincent Ng

Human Language Technology Research Institute

University of Texas at Dallas

Event Coreference Resolution

- Determines which event mentions in a text refer to the same real-world event

Event Coreference Resolution: Example

- Determines which event mentions in a text refer to the same real-world event

(John Cole) was cycling on (the road) (yesterday) and was [injured] when (two men) [stabbed] (him) with (a knife). (The police) are investigating (the mens)’ [criminal] motivation.

Event Coreference Resolution: Example

- Determines which event mentions in a text refer to the same real-world event

(John Cole) was cycling on (the road) (yesterday) and was [injured] when (two men) [stabbed] (him) with (a knife). (The police) are investigating (the mens)’ [criminal] motivation.

- Three event mentions: [injured], [stabbed], [criminal]
  - [stabbed] and [criminal] are coreferent because they refer to the same real-world event

What information do we need for event coreference?

(John Cole) was cycling on (the road) (yesterday) and was [injured] when (two men) [stabbed] (him) with (a knife). (The police) are investigating (the mens)' [criminal] motivation.

The word triggering the event mention

	Trigger Word	Event Type	Arguments of Event Mentions	Entity Coreference
E1	injured			
E2	stabbed			
E3	criminal			

What information do we need for event coreference?

(John Cole) was cycling on (the road) (yesterday) and was [injured] when (two men) [stabbed] (him) with (a knife). (The police) are investigating (the mens)' [criminal] motivation.

Coreferent event mentions must have the same event type

	Trigger Word	Event Type	Arguments of Event Mentions	Entity Coreference
E1	injured	Injury		
E2	stabbed	Attack		
E3	criminal	Attack		

What information do we need for event coreference?

(John Cole) was cycling on (the road) (yesterday) and was [injured] when (two men) [stabbed] (him) with (a knife). (The police) are investigating (the mens)' [criminal] motivation.

Coreferent mentions must have compatible arguments

	Trigger Word	Event Type	Arguments of Event Mentions	Entity Coreference
E1	injured	Injury	John Cole (Victim) the road (Location)….	
E2	stabbed	Attack	two men (Attacker) him (target) A knife (Instrument)	
E3	criminal	Attack	The mens (Attacker) John Cole (target)	

What information do we need for event coreference?

(John Cole) was cycling on (the road) (yesterday) and was [injured] when (two men) [stabbed] (him) with (a knife). (The police) are investigating (the mens)' [criminal] motivation.

To determine compatibility of two arguments

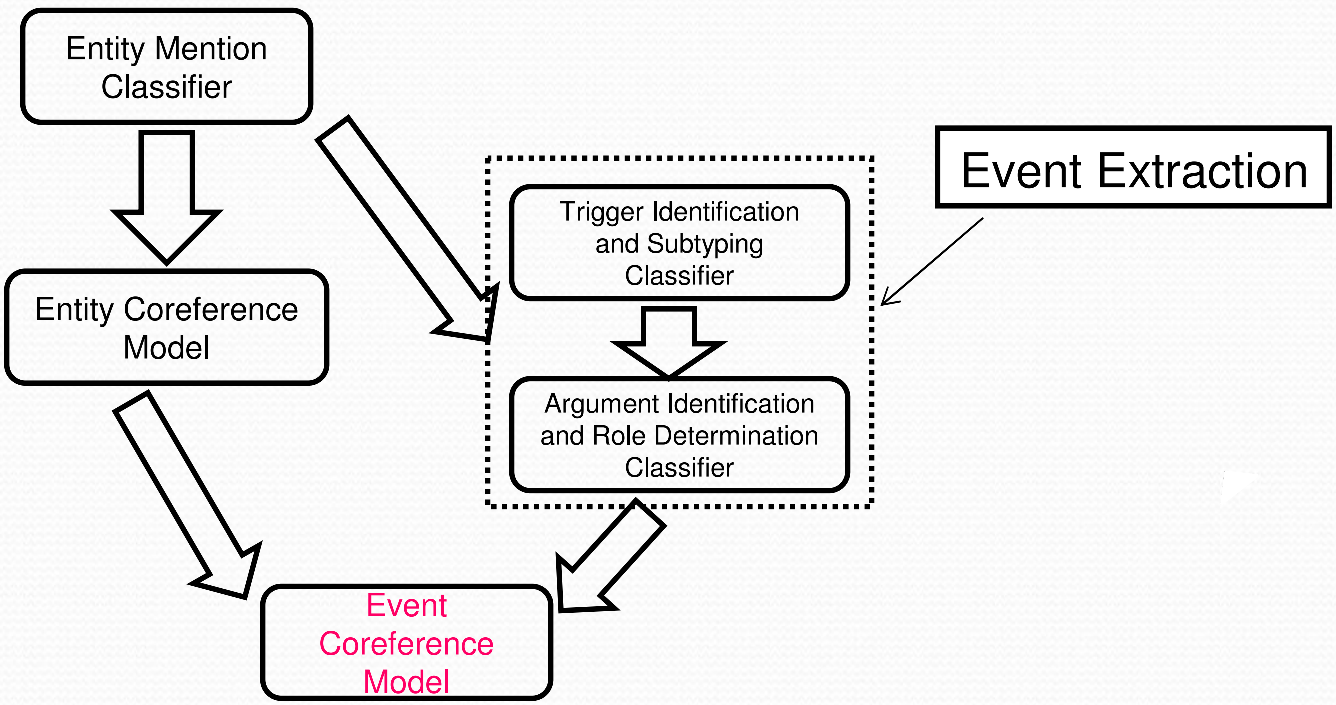
	Trigger Word	Event Type	Arguments of Event Mentions	Entity Coreference
E1	injured	Injury	John Cole (Victim) the road (Location)….	"two men" is coref with "The men" "him" is coref with "John Cole"
E2	stabbed	Attack	two men (Attacker) him (target) A knife (Instrument)	
E3	criminal	Attack	The mens (Attacker) John Cole (target)	

More Challenging than Entity Coreference

- An event coreference resolver lies at the end of information extraction pipeline
  - Rely on the noisy outputs produced by its upstream components

More Challenging than Entity Coreference

- An event coreference resolver lies at the end of information extraction pipeline
  - Rely on the noisy outputs produced by its upstream components



Goal

- Address a challenging version of this challenging task

End-to-end unsupervised Event Coreference Resolution

- Design an unsupervised event coreference model

Chinese Event Coreference Resolution

- Same as English event coreference in terms of task definition

- But… it has an additional challenge
  - Lack of large lexical resources such as FrameNet (Baker et al., 1998) and WordNet (Fellbaum, 1998) that have proven useful for English event coreference

Plan for the Talk

- Related work
- Unsupervised event coreference model
- Evaluation

Plan for the Talk

- Related work
- Unsupervised event coreference model
- Evaluation

Related Work

- Much work on event coreference resolution are for English
  - ACE corpus
  - ECB corpus
  - OntoNotes corpus
  - IC corpus

Related Work

- Much work on event coreference resolution are for English
  - ACE corpus
    - annotated events that belong to one of 33 event subtypes
    - Ahn (2006) and Chen and Ji (2009) apply supervised approach
    - the corpus we are using for evaluation
  - ECB corpus
  - OntoNotes corpus
  - IC corpus

Related Work

- Much work on event coreference resolution are for English
  - ACE corpus
  - ECB corpus
    - annotated mainly for cross-document event coreference, but many difficult cases of within-document event coreference links are not annotated (Liu et al., 2014)
    - Bejan and Harabagiu (2010; 2014) and Lee et al. (2012)
  - OntoNotes corpus
  - IC corpus

Related Work

- Much work on event coreference resolution are for English
  - ACE corpus
  - ECB corpus
  - OntoNotes corpus
    - not explicitly annotated with event coreference links
    - Chen et al. (2011) regard event coreference chains are all and only those coreference chains that involve at least one verb
  - IC corpus

Related Work

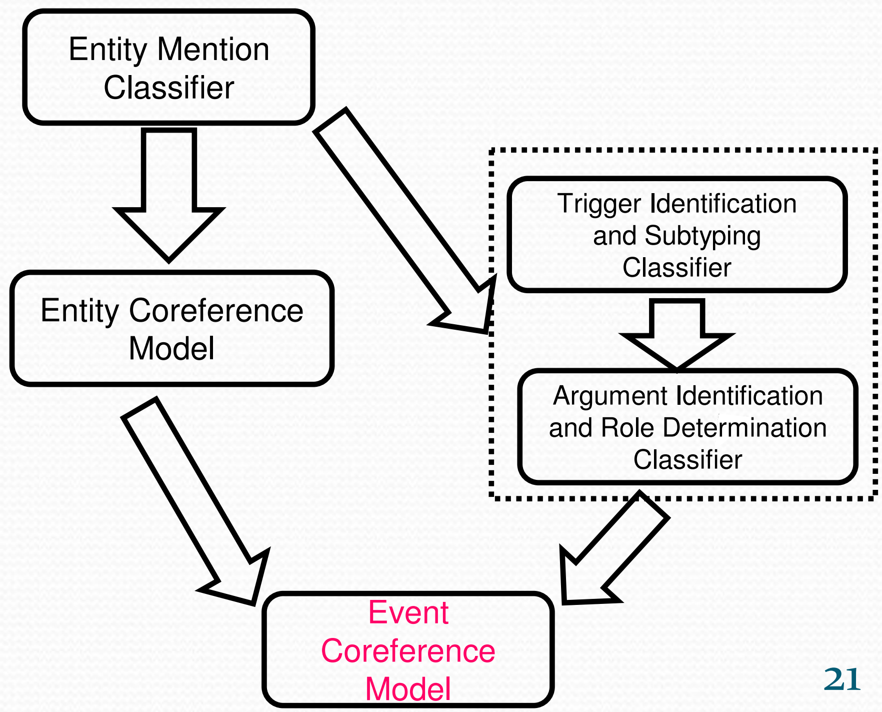
- Much work on event coreference resolution are for English
  - ACE corpus
  - ECB corpus
  - OntoNotes corpus
  - IC corpus
    - annotated not only full event coreference relations but also partial event coreference relations (Hovy et al., 2013)
    - Cybulska and Vossen (2012) and Goyal et al. (2013) exploit semantic relations and distributional semantic information

Related Work

- Much less work on Chinese event coreference resolution
  - SinoCoreferencer (Chen and Ng, 2014)
    - publicly available ACE-style supervised Chinese event coreference resolver that achieves state-of-the-art results

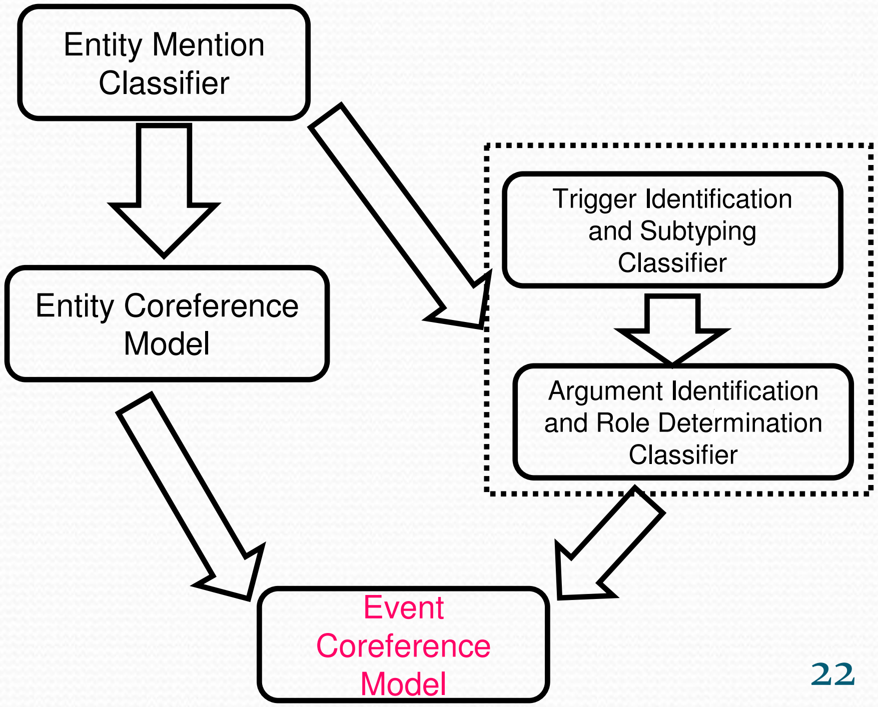
Related Work

- Much less work on Chinese event coreference resolution
  - SinoCoreferencer (Chen and Ng, 2014)
    - publicly available ACE-style supervised Chinese event coreference resolver that achieves state-of-the-art results
    - implements all of the IE components in the pipeline



Related Work

- Much less work on Chinese event coreference resolution
  - SinoCoreferencer (Chen and Ng, 2014)
    - publicly available ACE-style supervised Chinese event coreference resolver that achieves state-of-the-art results
    - implements all of the IE components in the pipeline
    - used as our baseline



Plan for the Talk

- Related Work
- Unsupervised Event Coreference Model
- Evaluation

If we had annotated training data …

- we could adopt the standard supervised approach:

If we had annotated training data …

- we could adopt the standard supervised approach:
  - Train a pairwise model to determine the probability that an event e and a candidate antecedent c given their context k are coreferent, i.e., P(coref=+|e,c,k)

If we had annotated training data …

- we could adopt the standard supervised approach:
  - Train a pairwise model to determine the probability that an event e and a candidate antecedent c given their context k are coreferent, i.e., P(coref=+|e,c,k)

(John Cole) was cycling on (the road) (yesterday) and was [injured] when (two men) [stabbed] (him) with (a knife). (The police) are investigating (the mens)' [criminal] motivation.

If we had annotated training data …

- we could adopt the standard supervised approach:
  - Train a pairwise model to determine the probability that an event e and a candidate antecedent c given their context k are coreferent, i.e., P(coref=+|e,c,k)

(John Cole) was cycling on (the road) (yesterday) and was [injured] when (two men) [stabbed] (him) with (a knife). (The police) are investigating (the mens)' [criminal] motivation.

Training Instances:

coref?	Event Mention	Candidate Antecedent
-	stabbed	injured
-	criminal	injured
+	criminal	stabbed

If we had annotated training data …

- we could adopt the standard supervised approach:
  - Train a pairwise model to determine the probability that an event e and a candidate antecedent c given their context k are coreferent, i.e., P(coref=+|e,c,k)
  - Apply the model to each event mention to select the candidate with the highest probability as its antecedent

But we don't have annotated data...

coref?	Event Mention	Candidate Antecedent
?	stabbed	injured
?	criminal	injured
?	criminal	stabbed

But we don't have annotated data…

coref?	Event Mention	Candidate Antecedent
0.2	stabbed	injured
0.4	criminal	injured
0.6	criminal	stabbed

- **Idea**: design a generative model and use EM to iteratively
  - Fill in missing values probabilistically (**E-step**)
    - i.e., determine the probability each pair of mentions is coreferent

But we don't have annotated data...

coref?	Event Mention	Candidate Antecedent
0.2	stabbed	injured
0.4	criminal	injured
0.6	criminal	stabbed

- **Idea**: design a generative model and use EM to iteratively
  - Fill in missing values probabilistically (**E-step**)
    - i.e., determine the probability each pair of mentions is coreferent
  - Estimate model parameters using the filled values (**M-step**)

But we don't have annotated data…

coref?	Event Mention	Candidate Antecedent
0.2	stabbed	injured
0.4	criminal	injured
0.6	criminal	stabbed

- We jointly perform two subtasks
  - Determine whether an event mention has an antecedent
  - If yes, find the antecedent

But we don't have annotated data…

coref?	Event Mention	Candidate Antecedent
0.2	stabbed	injured
0.4	criminal	injured
0.6	criminal	stabbed

- How to perform them jointly?
  - Introduce a *dummy* candidate antecedent for event mention

But we don't have annotated data…

coref?	Event Mention	Candidate Antecedent
0.2	stabbed	injured
0.3	criminal	injured
0.6	criminal	stabbed
0.5	stabbed	dummy
0.1	criminal	dummy

- How to perform them jointly?
  - Introduce a *dummy* candidate antecedent for event mention
  - If, for an event mention, the dummy has a higher probability than all other candidates, we posit it as **not** having an antecedent

Generative Model

- fill in the missing class values probabilistically
  - i.e., compute P(coref=+|e,c,k)

e: current event mention
c: event candidate antecedent
k: context for event coreference

Generative Model

- fill in the missing class values probabilistically
  - i.e., compute P(coref=+|e,c,k)

e: current event mention
c: event candidate antecedent
k: context for event coreference

- Using Chain Rule,

$$P(coref=+|e,c,k) = \frac{P(e,c,k,coref=+)}{Z}$$

- $Z = P(e,c,k)$ is a normalization constant

Generative Model

- fill in the missing class values probabilistically
  - i.e., compute P(coref=+|e,c,k)

e: current event mention
c: event candidate antecedent
k: context for event coreference

- Using Chain Rule,

$$P(coref = + \mid e, c, k) = \frac{P(e, c, k, coref = +)}{Z}$$

- Applying Chain Rule to the numerator,

$$P(e, c, k, coref = +)$$
$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Generative Model

- fill in the missing class values probabilistically
  - i.e., compute P(coref=+|e,c,k)

e: current event mention
c: event candidate antecedent
k: context for event coreference

- Using Chain Rule,

$$P(coref=+|e,c,k)=\frac{P(e,c,k,coref=+)}{Z}$$

- Applying Chain Rule to the numerator,

$$P(e,c,k,coref=+)$$
$$=P(k)P(c|k)P(coref=+|c,k)P(e|coref=+,c,k)$$

This is our generative model!

Generative Model

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e,c,k,coref=+)$$

$$=P(k)P(c\mid k)P(coref=+\mid c,k)P(e\mid coref=+,c,k)$$

Generative Model

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

generate
context k

Generative Model

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

generate
candidate c
given context k

Generative Model

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

generate class label
given candidate c
and context k

Generative Model

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

generate event mention e given class label, candidate c and context k

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

These four are the **model parameters**

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$
$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Assumption: for each event mention, the contexts generated from different candidate antecedents have the same probability

- Effectively ignoring this term

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Prior probability of a candidate antecedent c given context k

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Prior probability of a candidate antecedent c given context k

How to estimate this probability?

- If the candidate c have different event type as e, we set the prior to 0
- Uniform distribution for all the other candidates

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$
$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Prior probability that they are coreferent given candidate & context

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Prior probability that they are coreferent given candidate & context

How to estimate this probability?

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Prior probability that they are coreferent given candidate & context

How to estimate this probability?

- represent context k using 6 features (more on it later)

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Prior probability that they are coreferent given candidate & context

How to estimate this probability?

- represent context k using 6 features (more on it later)
  - estimate probability in the M-step

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$
$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Probability of e given everything else

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Probability of e given everything else

How to estimate $P(e \mid coref = +, c, k)$ **?**

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Probability of e given everything else

How to estimate $P(e \mid coref = +, c, k)$ **?**

- simplify by dropping k, yielding

$$P(e \mid coref = +, c)$$

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Probability of e given everything else

How to estimate $P(e \mid coref = +, c, k)$ **?**

- simplify by dropping k, yielding

$$P(e \mid coref = +, c)$$

- approximate e and c by their triggers' word, yielding

$$P(e_t \mid coref = +, c_t)$$

How to estimate each of these parameters?

e: current event mention
c: event candidate antecedent
k: context for event coreference

$$P(e, c, k, coref = +)$$

$$= P(k)P(c \mid k)P(coref = + \mid c, k)P(e \mid coref = +, c, k)$$

Probability of e given everything else

How to estimate $P(e \mid coref = +, c, k)$ **?**

- simplify by dropping k, yielding

$$P(e \mid coref = +, c)$$

- approximate e and c by their triggers' word, yielding

$$P(e_t \mid coref = +, c_t)$$

- estimate $P(e_t \mid coref = +, c_t)$ in the M-step

Six Features for Representing Two Event Mentions and their Contexts

- Feature 1: encodes three coreference conditions
  - Determine whether their triggers satisfy any of the following conditions:
    - are lexically identical
    - contain same basic verb and have compatible verb structures
    - their word2vec similarity exceeds 0.8

Six Features for Representing Two Event Mentions and their Contexts

- Feature 1: encodes three coreference conditions

- Feature 2-5: encode non-coreference conditions
  - whether they are incompatible w.r.t. number
  - whether they possess two arguments that have the same semantic role but different semantic classes
  - whether they possess two arguments that have the same semantic role but are not coreferent
  - whether they possess two different values as their arguments

Six Features for Representing Two Event Mentions and their Contexts

- Feature 1: encodes three coreference conditions
- Feature 2-5: encode non-coreference conditions
- Feature 6: distance feature
  - encodes their distance in terms of the number of separating event mentions

Features for Dummy Candidates

- How to compute these six features for a *dummy* candidate?

Features for Dummy Candidates

- How to compute these six features for a *dummy* candidate?
  - For feature 1 (coreference condition)
    - we set the feature value of *dummy* to True,

Features for Dummy Candidates

- How to compute these six features for a *dummy* candidate?
  - For feature 1 (coreference condition)
    - we set the feature value of *dummy* to True,
  - For features 2-5 (non-coreference conditions)
    - we set the feature value of *dummy* to False

Features for Dummy Candidates

- How to compute these six features for a *dummy* candidate?
  - For feature 1 (coreference condition)
    - we set the feature value of *dummy* to True,
  - For features 2-5 (non-coreference conditions)
    - we set the feature value of *dummy* to False
  - Feature 6 (distance)
    - Assume it is the 0^{th} event mention in the document

Features for Dummy Candidates

- How to compute these six features for a *dummy* candidate?
  - For feature 1 (coreference condition)
    - we set the feature value of *dummy* to True,
  - For features 2-5 (non-coreference conditions)
    - we set the feature value of *dummy* to False
  - Feature 6 (distance)
    - Assume it is the 0^{th} event mention in the document

Increase the likelihood that the dummy is chosen as the antecedent

Features for Dummy Candidates

- How to compute these six features for a *dummy* candidate?
  - For feature 1 (coreference condition)
    - we set the feature value of *dummy* to True,
  - For features 2-5 (non-coreference conditions)
    - we set the feature value of *dummy* to False
  - Feature 6 (distance)
    - Assume it is the 0^{th} event mention in the document

Increase the likelihood that the dummy is chosen as the antecedent

Decrease the likelihood that the dummy is chosen as the antecedent

Features for Dummy Candidates

- How to compute these six features for a *dummy* candidate?
  - For feature 1 (coreference condition)
    - we set the feature value of *dummy* to True,
  - For features 2-5 (non-coreference conditions)
    - we set the feature value of *dummy* to False
  - Feature 6 (distance)
    - Assume it is the 0th event mention in the document

Increase the likelihood that the dummy is chosen as the antecedent

Decrease the likelihood that the dummy is chosen as the antecedent

If the event mention to be resolved is non-anaphoric: Features 1-5 will make the dummy more likely to be chosen as the the antecedent

Features for Dummy Candidates

- How to compute these six features for a *dummy* candidate?
  - For feature 1 (coreference condition)
    - we set the feature value of *dummy* to True,
  - For features 2-5 (non-coreference conditions)
    - we set the feature value of *dummy* to False
  - Feature 6 (distance)
    - Assume it is the 0^{th} event mention in the document

Increase the likelihood that the dummy is chosen as the antecedent

Decrease the likelihood that the dummy is chosen as the antecedent

If the event mention to be resolved is anaphoric: Feature 6 will make the dummy less likely to be chosen as the the antecedent

The EM Algorithm: Recap

- E-step:
  - Fill in the missing class values probabilistically by computing P(coref=+|e,c,k) using the current model parameter values
- M-step:
  - Re-estimate the model parameters using maximum likelihood estimation
- We start in the M-step by initializing all the parameters to uniform values and run EM until convergence

Applying the Learned Model to Test Data

- Use the model to compute the probability that each event mention e is coreferent with each candidate antecedent c
- For each e, pick c with highest probability as its antecedent
  - If c is the *dummy* candidate antecedent, then posit e as non-anaphoric

Plan for the Talk

- Related work
- Unsupervised event coreference model
- Evaluation

Experimental Setup

- Corpus
  - Five-fold cross validation on Chinese portion of ACE 2005 training corpus

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

- Evaluation measures
  - MUC, B^3, $CEAF_e$ and BLANC
  - CoNLL score: unweighted average of the MUC, B^3, and $CEAF_e$ F-scores

Evaluation Setting

- End-to-end evaluation
  - SinoCoreferencer is used to provide entity extraction, entity coreference and event extraction outputs as inputs for our event coreference model
- 5-fold cross validation

Two Supervised Baseline Systems

- Rote learning
  - posits two event mentions as coreferent if their corresponding triggers are annotated as coreferent in the training data

- SinoCoreferencer
  - state-of-the-art supervised Chinese event coreference resolver

Results: Rote Learning Baseline

	MUC			B^3			$CEAF_e$			BLANC			Avg
System	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

- Rote learning baseline achieves a CoNLL score of 37.9

Results: SinoCoreferencer Baseline

	MUC			B^3			$CEAF_e$			BLANC			Avg
System	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
SinoCorefencer	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

- Rote learning baseline achieves a CoNLL score of 37.9
- SinoCoreferencer outperforms rote learning baseline

Results: Our Unsupervised Model

	MUC			B^3			$CEAF_e$			BLANC			Avg
System	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
SinoCorefencer	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**

- The rote learning baseline achieves a CoNLL score of 37.9
- The SinoCoreferencer outperforms the rote learning baseline
- Our model outperforms both baseline systems, achieving a CoNLL score of 41.5

Ablation Experiments

- In each experiment, remove exactly one probability term or feature from our model and retrain the model

Ablation Experiments: Results

System	MUC	B3	CEAFe	BLANC	AVG
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

Ablation Experiments: Results

System	MUC	B3	CEAFe	BLANC	AVG
Full Model	42.8	40.2	41.6	26.9	41.5
-$P(e_t \mid c_t)$	42.9	39.8	40.9	26.9	41.2
-$P(c \mid k)$	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

- Feature 1 (coreference conditions) is most useful

Ablation Experiments: Results

System	MUC	B3	CEAFe	BLANC	AVG
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

- Feature 1 (coreference conditions) is most useful
- P(c|k) (probability of a candidate antecedent given context) is the second most useful term

Summary

- Proposed an unsupervised model for Chinese event coreference resolution
  - rivaled its supervised counterparts in performance when evaluated on the Chinese portion of the ACE 2005 training data