Chinese Zero Pronoun Resolution: An Unsupervised Probabilistic Model Rivaling Supervised Resolvers

Chen Chen and **Vincent Ng**
Human Language Technology Research Institute
University of Texas at Dallas

Zero Pronouns

- A zero pronoun (ZP) is a gap in a sentence
  - found when a phonetically null form is used to refer to an entity

Zero Pronouns

- A zero pronoun (ZP) is a gap in a sentence
  - found when a phonetically null form is used to refer to an entity

- An anaphoric zero pronoun (AZP) is a ZP that is anaphoric
  - a ZP that corefers with one or more preceding NPs in the text

Zero Pronouns

- A zero pronoun (ZP) is a gap in a sentence
  - found when a phonetically null form is used to refer to an entity

- An anaphoric zero pronoun (AZP) is a ZP that is anaphoric
  - a ZP that corefers with one or more preceding NPs in the text

俄罗斯作为米洛舍夫维奇一贯的支持者，
*pro*曾经提出调停这场政治危机。

Russia is a consistent support of Milosevic,
pro has proposed to mediate the political crisis.

Zero Pronoun Resolution

- More challenging than overt pronoun resolution
  - ZPs lack grammatical attributes useful for overt pronoun resolution such as gender and number

Zero Pronoun Resolution

- More challenging than overt pronoun resolution
  - ZPs lack grammatical attributes useful for overt pronoun resolution such as gender and number

- Typically composed of two steps
  - **AZP identification**
    - Extract from a text all the ZPs that are anaphoric
  - **AZP resolution**
    - Identify an antecedent of an AZP

Zero Pronoun Resolution

- More challenging than overt pronoun resolution
  - ZPs lack grammatical attributes useful for overt pronoun resolution such as gender and number

- Typically composed of two steps
  - **AZP identification**
    - Extract from a text all the ZPs that are anaphoric
  - **AZP resolution**
    - Identify an antecedent of an AZP

- State of the art resolvers: supervised approach
  - Train one classifier for AZP identification and another one for AZP resolution

Our Focus: AZP Resolution

- Design a model that assumes as input the AZPs in a document and resolves each of them

Our Focus: AZP Resolution

- Design a model that assumes as input the AZPs in a document and resolves each of them

- AZP resolution alone is by no means easy
  - State-of-the-art supervised resolvers achieved an F-score of only 47.7% for resolving manually identified Chinese AZPs

Our Focus: AZP Resolution

- Design a model that assumes as input the AZPs in a document and resolves each of them

- AZP resolution alone is by no means easy
  - State-of-the-art supervised resolvers achieved an F-score of only 47.7% for resolving manually identified Chinese AZPs

- Will evaluate our resolution model on both manually identified and automatically identified AZPs

Contribution

- An unsupervised model for AZP resolution that rivals its supervised counterparts in performance when evaluated on the Chinese portion of the OntoNotes v5.0 corpus

Contribution

- An unsupervised model for AZP resolution that rivals its supervised counterparts in performance when evaluated on the Chinese portion of the OntoNotes v5.0 corpus
  - Does not require training data with manually resolved AZPs

Contribution

- An unsupervised model for AZP resolution that rivals its supervised counterparts in performance when evaluated on the Chinese portion of the OntoNotes v5.0 corpus
  - Does not require training data with manually resolved AZPs
  - Underlying generative process is not language-dependent
    - Can be applied to languages without such annotated data

Contribution

- An unsupervised model for AZP resolution that rivals its supervised counterparts in performance when evaluated on the Chinese portion of the OntoNotes v5.0 corpus
  - Does not require training data with manually resolved AZPs
  - Underlying generative process is not language-dependent
    - Can be applied to languages without such annotated data
  - Based on a **novel hypothesis**

We can apply a probabilistic pronoun resolution model trained on overt pronouns in an unsupervised manner to resolve AZPs

Plan for the Talk

- Intro to Chinese overt pronouns
- Generative model for overt pronoun resolution
  - Training and application
- Evaluation

Plan for the Talk

- Intro to Chinese overt pronouns
- Generative model for overt pronoun resolution
  - Training and application
- Evaluation

Chinese Overt Pronouns

Pronoun	Number	Gender	Person	Animacy
我 (I)	singular	neuter	first	animate
你 (you)	singular	neuter	second	animate
他 (he)	singular	masculine	third	animate
她 (she)	singular	feminine	third	animate
它 (it)	singular	neuter	third	inanimate
你们 (you)	plural	neuter	second	animate
我们 (we)	plural	neuter	first	animate
他们 (they)	plural	masculine	third	animate
她们 (they)	plural	feminine	third	animate
它们 (they)	plural	neuter	third	inanimate

Chinese Overt Pronouns

Pronoun	Number	Gender	Person	Animacy
我 (I)	singular	neuter	first	animate
你 (you)	singular	neuter	second	animate
他 (he)	singular	masculine	third	animate
她 (she)	singular	feminine	third	animate
它 (it)	singular	neuter	third	inanimate
你们 (you)	plural	neuter	second	animate
我们 (we)	plural	neuter	first	animate
他们 (they)	plural	masculine	third	animate
她们 (they)	plural	feminine	third	animate
它们 (they)	plural	neuter	third	inanimate

- Use 10 overt pronouns
  - Each can be uniquely identified using these 4 attributes

Chinese Overt Pronouns

Pronoun	Number	Gender	Person	Animacy
我 (I)	singular	neuter	first	animate
你 (you)	singular	neuter	second	animate
他 (he)	singular	masculine	third	animate
她 (she)	singular	feminine	third	animate
它 (it)	singular	neuter	third	inanimate
你们 (you)	plural	neuter	second	animate
我们 (we)	plural	neuter	first	animate
他们 (they)	plural	masculine	third	animate
她们 (they)	plural	feminine	third	animate
它们 (they)	plural	neuter	third	inanimate

- Use 10 overt pronouns
  - Each can be uniquely identified using these 4 attributes

Plan for the Talk

- Intro to Chinese overt pronouns
- Generative model for overt pronoun resolution
  - Training and application
- Evaluation

Generative Model for Unsupervised Overt Pronoun Resolution

- estimates P(p, c, k, l)

Generative Model for Unsupervised Overt Pronoun Resolution

- estimates P(p, c, k, l)

overt pronoun to
be resolved

Generative Model for Unsupervised Overt Pronoun Resolution

- estimates P(p, c, k, l)

overt pronoun to
be resolved

a candidate
antecedent of p

Generative Model for Unsupervised Overt Pronoun Resolution

- estimates P(p, c, k, l)

overt pronoun to be resolved

a candidate antecedent of p

context surrounding p and **all** of its candidate antecedents (global context)

Generative Model for Unsupervised Overt Pronoun Resolution

- estimates P(p, c, k, l)

a binary variable indicating whether c is p's correct antecedent

overt pronoun to be resolved

a candidate antecedent of p

context surrounding p and **all** of its candidate antecedents (global context)

Generative Model for Unsupervised Overt Pronoun Resolution

- estimates P(p, c, k, l)

a binary variable indicating whether c is p's correct antecedent

overt pronoun to be resolved

a candidate antecedent of p

context surrounding p and **all** of its candidate antecedents (global context)

- This is an unsupervised procedure
  - probability estimated from an unannotated corpus
  - treat p, c, k as observed data and l as hidden data
  - use EM to estimate the model parameters

The EM Algorithm

- E-step
  - using current parameter estimates, label each overt pronoun p with the probability it co-refers with each candidate antecedent c given context k

$$P(l = 1 | p, c, k)$$

The EM Algorithm

- E-step
  - using current parameter estimates, label each overt pronoun p with the probability it co-refers with each candidate antecedent c given context k

$$P(l=1 \mid p,c,k)$$

p1	c1	?
p1	c2	?
p2	c1	?
p2	c2	?
p2	c3	?

Initially, we don't know what's the probability that a pronoun and a candidate antecedent are coreferent

The EM Algorithm

- E-step
  - using current parameter estimates, label each overt pronoun p with the probability it co-refers with each candidate antecedent c given context k

$$P(l = 1 | p, c, k)$$

p1	c1	0.87
p1	c2	0.11
p2	c1	0.42
p2	c2	0.98
p2	c3	0.69

E-step: fill in the missing value (the expected value of l) for each pair of pronouns and candidates

The EM Algorithm

- E-step
  - using current parameter estimates, label each overt pronoun p with the probability it co-refers with each candidate antecedent c given context k

$$P(l = 1 | p, c, k)$$

- M-step
  - (re)estimate model parameters from data containing the overt pronouns probabilistically labeled in the E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- **Goal**: estimate $P(l = 1 | p, c, k)$

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1|p,c,k)$

$$P(l=1|p,c,k)=\frac{P(p,c,k,l=1)}{P(p,c,k)}$$

Definition of conditional prob.

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1|p,c,k)$

$$P(l=1|p,c,k)=\frac{P(p,c,k,l=1)}{P(p,c,k)}$$

- Rewriting the denominator,

$$P(p,c,k)$$
$$=P(p,c,k,l=1)+P(p,c,k,l=0)$$

The two cases are mutually exclusive

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- **Goal**: estimate $P(l=1|p,c,k)$

$$P(l=1|p,c,k)=\frac{P(p,c,k,l=1)}{P(p,c,k)}$$

- Rewriting the denominator,

$$P(p,c,k)$$
$$=P(p,c,k,l=1)+P(p,c,k,l=0)$$

Assume **exactly** one of p's candidate antecedent is its correct antecedent

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- **Goal**: estimate $P(l=1|p,c,k)$

$$P(l=1|p,c,k)=\frac{P(p,c,k,l=1)}{P(p,c,k)}$$

- Rewriting the denominator,

$$P(p,c,k)$$
$$=P(p,c,k,l=1)+P(p,c,k,l=0)$$

Assume **exactly** one of p's candidate antecedent is its correct antecedent

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1|p,c,k)$

$$P(l=1|p,c,k)=\frac{P(p,c,k,l=1)}{P(p,c,k)}$$

- Rewriting the denominator,

Assume **exactly** one of p's candidate antecedent is its correct antecedent

$$\begin{aligned}&P(p,c,k)\\&=P(p,c,k,l=1)+P(p,c,k,l=0)\\&=P(p,c,k,l=1)+P(p,c_1,k,l=1)+\ldots+P(p,c_n,k,l=1)\end{aligned}$$

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- **Goal**: estimate $P(l=1 | p,c,k)$

$$P(l=1 | p,c,k) = \frac{P(p,c,k,l=1)}{P(p,c,k)}$$

Refers to the case where p and c are not coreferent (l=0)

- Rewriting the denominator,

$$\begin{aligned} &P(p,c,k) \\ &= P(p,c,k,l=1) + P(p,c,k,l=0) \\ &= P(p,c,k,l=1) + P(p,c_1,k,l=1) + \ldots + P(p,c_{n,}k,l=1) \end{aligned}$$

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1 | p,c,k)$

$$P(l=1 | p,c,k) = \frac{P(p,c,k,l=1)}{P(p,c,k)}$$

- Rewriting the denominator,

By assumption, one of the remaining candidates of p must be its correct antecedent

$$\begin{aligned} & P(p,c,k) \\ & = P(p,c,k,l=1) + P(p,c,k,l=0) \\ & = P(p,c,k,l=1) + P(p,c_1,k,l=1) + \ldots + P(p,c_{n,}k,l=1) \end{aligned}$$

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1|p,c,k)$

$$P(l=1|p,c,k)=\frac{P(p,c,k,l=1)}{P(p,c,k)}$$

Enumerates over all possible candidates

- Rewriting the denominator,

$$\begin{aligned}&P(p,c,k)\\&=P(p,c,k,l=1)+P(p,c,k,l=0)\\&=P(p,c,k,l=1)+P(p,c_1,k,l=1)+\ldots+P(p,c_n,k,l=1)\end{aligned}$$

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- **Goal**: estimate $P(l=1|p,c,k)$

$$P(l=1|p,c,k)=\frac{P(p,c,k,l=1)}{P(p,c,k)}$$

- Rewriting the denominator,

$$\begin{aligned}
&P(p,c,k)\\
&=P(p,c,k,l=1)+P(p,c,k,l=0)\\
&=P(p,c,k,l=1)+P(p,c_1,k,l=1)+\ldots+P(p,c_{n,}k,l=1)\\
&=\sum_{c'\in C}P(p,c',k,l=1)
\end{aligned}$$

Shorthand by using summation
Sum over all possible candidates

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- **Goal**: estimate $P(l=1|p,c,k)$

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

- Rewriting the denominator,

$$\begin{aligned}&P(p,c,k)\\&=P(p,c,k,l=1)+P(p,c,k,l=0)\\&=P(p,c,k,l=1)+P(p,c_1,k,l=1)+\ldots+P(p,c_{n,}k,l=1)\\&=\sum_{c'\in C}P(p,c',k,l=1)\end{aligned}$$

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1 \mid p,c,k)$

$$P(l=1 \mid p,c,k) = \frac{P(p,c,k,l=1)}{P(p,c,k)} = \frac{P(p,c,k,l=1)}{\sum_{c' \in C} P(p,c',k,l=1)}$$

These two probabilities have the same form

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1 | p,c,k)$

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

- Using Chain Rule, we can decompose

$$P(p,c,k,l=1)$$
$$= P(p | k,c,l=1)\, P(l=1 | k,c)\, P(c | k)\, P(k)$$

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1|p,c,k)$

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

- Using Chain Rule, we can decompose

$$P(p,c,k,l=1)$$
$$=P(p|k,c,l=1)\,P(l=1|k,c)\,P(c|k)\,P(k)$$

generate
context k

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1 \mid p,c,k)$

$$P(l=1 \mid p,c,k) = \frac{P(p,c,k,l=1)}{P(p,c,k)} = \frac{P(p,c,k,l=1)}{\sum_{c' \in C} P(p,c',k,l=1)}$$

- Using Chain Rule, we can decompose

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$

generate candidate antecedent c given context k

generate context k

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1|p,c,k)$

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

- Using Chain Rule, we can decompose

$$P(p,c,k,l=1)$$
$$=P(p|k,c,l=1)\,\underbrace{P(l=1|k,c)}_{\text{generate coref label l given candidate c and context k}}\,\underbrace{P(c|k)}_{\text{generate candidate antecedent c given context k}}\,\underbrace{P(k)}_{\text{generate context k}}$$

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1|p,c,k)$

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

- Using Chain Rule, we can decompose

$$P(p,c,k,l=1)$$
$$=P(p|k,c,l=1)\,P(l=1|k,c)\,P(c|k)\,P(k)$$

generate pronoun p given coref label l, candidate c and context k

generate coref label l given candidate c and context k

generate candidate antecedent c given context k

generate context k

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- Using Chain Rule, we can decompose

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1)\,P(l=1 \mid k,c)\,P(c \mid k)\,P(k)$$

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Using Chain Rule, we can decompose

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1) P(l=1 \mid k,c) P(c \mid k) P(k)$$

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- Using Chain Rule

$$P(p,c,k,l=1)$$

$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$

Probability of generating p given k and c, and that p and c are coreferent

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$=P(p\,|\,k,c,l=1)\,P(l=1\,|\,k,c)\,P(c\,|\,k)\,P(k)$$

$$P(p\,|\,k,c,l=1)\approx P(p\,|\,c,l=1)$$

rewrite by dropping k

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$

$$P(p \mid k,c,l=1) \approx P(p \mid c,l=1)$$

Reason:

given l=1, we assume we can generate p from c without k

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$

$$P(p \mid k,c,l=1) \approx P(p \mid c,l=1)$$

the man → he
the women → they

Reason:

given l=1, we assume we can generate p from c without k

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$

$$P(p \mid k,c,l=1) \approx \boxed{P(p \mid c,l=1)}$$
$$\approx \boxed{P(p_{Num}, p_{Gen}, p_{Per}, p_{Ani} \mid c,l=1)}$$

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1)\,P(l=1 \mid k,c)\,P(c \mid k)\,P(k)$$

$$P(p \mid k,c,l=1) \approx P(p \mid c,l=1)$$
$$\approx P(p_{Num}, p_{Gen}, p_{Per}, p_{Ani} \mid c,l=1)$$

Represent p using its four grammatical attribute values: Num, Gen, Per, Ani

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Using Chain Rule

$$P(p, c, k, l = 1)$$
$$= P(p \mid k, c, l = 1) P(l = 1 \mid k, c) P(c \mid k) P(k)$$

$$P(p \mid k, c, l = 1) \approx P(p \mid c, l = 1)$$
$$\approx P(p_{Num}, p_{Gen}, p_{Per}, p_{Ani} \mid c, l = 1)$$
$$\approx P(p_{Num} \mid c_{Num}, l = 1) P(p_{Gen} \mid c_{Gen}, l = 1) P(p_{Per} \mid c_{Per}, l = 1) P(p_{Ani} \mid c_{Ani}, l = 1)$$

decompose the joint probability into 4 smaller probabilities

E-step

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

- Using Chain Rule

$$P(p,c,k,l=1)$$

$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$

$$P(p \mid k,c,l=1) \approx P(p \mid c,l=1)$$

$$\approx P(p_{Num}, p_{Gen}, p_{Per}, p_{Ani} \mid c, l=1)$$

$$\approx P(p_{Num} \mid c_{Num}, l=1) P(p_{Gen} \mid c_{Gen}, l=1)$$
$$P(p_{Per} \mid c_{Per}, l=1) P(p_{Ani} \mid c_{Ani}, l=1)$$

assume p's value w.r.t. attribute a can be generated independently of other values given c's value w.r.t. a

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$
$$= \prod_{a \in A} P(p_a \mid c_a, l=1) P(l=1 \mid k,c) P(c \mid k) P(k)$$

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$= P(p\,|\,k,c,l=1)\,P(l=1\,|\,k,c)\,P(c\,|\,k)\,P(k)$$
$$= \prod_{a\in A} P(p_a\,|\,c_a,l=1)\,P(l=1\,|\,k,c)P(c\,|\,k)P(k)$$

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

E-step

- Using Chain Rule

$$P(p,c,k,l=1)$$

$$= P(p\,|\,k,c,l=1)\,P(l=1\,|\,k,c)\,P(c\,|\,k)\,P(k)$$

$$= \prod_{a\in A} P(p_a\,|\,c_a,l=1)P(l=1\,|\,k,c)P(c\,|\,k)P(k)$$

Probability that p and c are coreferent given k and c

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1)\,P(l=1 \mid k,c)\,P(c \mid k)\,P(k)$$
$$= \prod_{a \in A} P(p_a \mid c_a, l=1)\,P(l=1 \mid k,c)\,P(c \mid k)\,P(k)$$

$$P(l=1 \mid k,c) \approx P(l=1 \mid k_c, c)$$

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- Using Chain Rule

$$
\begin{aligned}
&P(p,c,k,l=1) \\
&= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k) \\
&= \prod_{a \in A} P(p_a \mid c_a, l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)
\end{aligned}
$$

$$P(l=1 \mid k,c) \approx P(l=1 \mid k_c, c)$$

Replace k with k_c

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

E-step

- Using Chain Rule

$$P(p,c,k,l=1)$$

$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$

$$= \prod_{a \in A} P(p_a \mid c_a, l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$

$$P(l=1 \mid k,c) \approx P(l=1 \mid k_c, c)$$

Replace k with k_c

k can be thought of as the global context surrounding p and all of its candidates

k_c can be thought of as the local context surrounding p and candidate c

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Using Chain Rule

$$P(p,c,k,l=1)$$

$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$

$$= \prod_{a \in A} P(p_a \mid c_a, l=1) P(l=1 \mid k,c) P(c \mid k) P(k)$$

$$P(l=1 \mid k, c) \approx P(l=1 \mid k_c, c)$$

given k_c (**local context** surrounding p and c), whether p and c are coreferent is not affected by remaining context

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Using Chain Rule

$$P(p, c, k, l=1)$$

$$= P(p \mid k, c, l=1)\, P(l=1 \mid k, c)\, P(c \mid k)\, P(k)$$

$$= \prod_{a \in A} P(p_a \mid c_a, l=1)\, P(l=1 \mid k, c)\, P(c \mid k)\, P(k)$$

$$P(l=1 \mid k, c) \approx P(l=1 \mid k_c, c) \approx P(l=1 \mid k_c)$$

Rewrite by dropping c

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$=P(p\,|\,k,c,l=1)\,P(l=1\,|\,k,c)\,P(c\,|\,k)\,P(k)$$
$$=\prod_{a\in A}P(p_a\,|\,c_a,l=1)P(l=1\,|\,k,c)P(c\,|\,k)P(k)$$

$$P(l=1\,|\,k,c)\approx P(l=1\,|\,k_c,c)\approx P(l=1\,|\,k_c)$$

Rewrite by dropping c

local context k_c is sufficient for determining whether p and c are coreferent

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$
$$= \prod_{a\in A} P(p_a \mid c_a, l=1) P(l=1 \mid k,c) P(c \mid k) P(k)$$
$$= \prod_{a\in A} P(p_a \mid c_a, l=1) P(l=1 \mid k_c) P(c \mid k) P(k)$$

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

E-step

- Using Chain Rule

$$
\begin{aligned}
& P(p,c,k,l=1) \\
&= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k) \\
&= \prod_{a \in A} P(p_a \mid c_a, l=1) P(l=1 \mid k,c) P(c \mid k) P(k) \\
&= \prod_{a \in A} P(p_a \mid c_a, l=1) P(l=1 \mid k_c) P(c \mid k) P(k)
\end{aligned}
$$

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1)\,P(l=1 \mid k,c)\,P(c \mid k)\,P(k)$$
$$= \prod_{a \in A} P(p_a \mid c_a, l=1) P(l=1 \mid k,c) P(c \mid k) P(k)$$
$$= \prod_{a \in A} P(p_a \mid c_a, l=1) P(l=1 \mid k_c) P(c \mid k) P(k)$$

Probability of
c given k

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Using Chain Rule

$$P(p,c,k,l=1)$$
$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$
$$= \prod_{a \in A} P(p_a \mid c_a, l=1) P(l=1 \mid k,c) P(c \mid k) P(k)$$
$$= \prod_{a \in A} P(p_a \mid c_a, l=1) P(l=1 \mid k_c) P(c \mid k) P(k)$$

$$P(c \mid k) \approx P(c' \mid k) \quad \forall c, c' \in C$$

E-step

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

- Using Chain Rule

$$P(p,c,k,l=1)$$

$$= P(p \mid k,c,l=1)\, P(l=1 \mid k,c)\, P(c \mid k)\, P(k)$$

$$= \prod_{a \in A} P(p_a \mid c_a, l=1) P(l=1 \mid k,c) P(c \mid k) P(k)$$

$$= \prod_{a \in A} P(p_a \mid c_a, l=1) P(l=1 \mid k_c) P(c \mid k) P(k)$$

$$P(c \mid k) \approx P(c' \mid k) \quad \forall c, c' \in C$$

given context k, all candidate antecedents of p can be generated with equal probability

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- Goal: estimate $P(l=1|p,c,k)$

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

We have been trying to decompose this joint probability

E-step

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Goal: estimate $P(l=1|p,c,k)$

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

Using the assumptions we have made so far, we can rewrite this probability

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

E-step

- Goal: estimate $P(l=1|p,c,k)$

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

$$=\frac{\prod_{a\in A}P(p_a|c_a,l=1)P(l=1|k_c)}{\sum_{c'\in C}\prod_{a\in A}P(p_a|c'_a,l=1)P(l=1|k_{c'})}$$

E-step

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

- Goal: estimate $P(l=1|p,c,k)$

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

$$=\frac{\prod_{a\in A}P(p_a|c_a,l=1)P(l=1|k_c)}{\sum_{c'\in C}\prod_{a\in A}P(p_a|c'_a,l=1)P(l=1|k_{c'})}$$

Two sets of model parameters

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

E-step

- Goal: estimate $P(l=1|p,c,k)$

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

$$=\frac{\prod_{a\in A}P(p_a|c_a,l=1)P(l=1|k_c)}{\sum_{c'\in C}\prod_{a\in A}P(p_a|c'_a,l=1)P(l=1|k_{c'})}$$

First set of parameters

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

E-step

- Goal: estimate $P(l=1|p,c,k)$

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

$$=\frac{\prod_{a\in A}P(p_a|c_a,l=1)P(l=1|k_c)}{\sum_{c'\in C}\prod_{a\in A}P(p_a|c'_a,l=1)P(l=1|k_{c'})}$$

First set of parameters

given p and c are coreferent, the probability of generating p's attribute values from c's attribute values

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

E-step

- Goal: estimate $P(l=1|p,c,k)$

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

$$=\frac{\prod_{a\in A}P(p_a|c_a,l=1)P(l=1|k_c)}{\sum_{c'\in C}\prod_{a\in A}P(p_a|c'_a,l=1)P(l=1|k_{c'})}$$

First set of parameters

given p and c are coreferent, the probability of generating p's attribute values from c's attribute values

Second set of parameters

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

E-step

- **Goal**: estimate $P(l=1 \mid p,c,k)$

$$P(l=1 \mid p,c,k) = \frac{P(p,c,k,l=1)}{P(p,c,k)} = \frac{P(p,c,k,l=1)}{\sum_{c' \in C} P(p,c',k,l=1)}$$

$$= \frac{\prod_{a \in A} P(p_a \mid c_a, l=1) P(l=1 \mid k_c)}{\sum_{c' \in C} \prod_{a \in A} P(p_a \mid c'_a, l=1) P(l=1 \mid k_{c'})}$$

First set of parameters

given p and c are coreferent, the probability of generating p's attribute values from c's attribute values

Second set of parameters

given the local context surrounding p and k, the probability that p and c are coref

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

M-step

- **Goal**: given $P(l=1|p,c,k)$, estimate model parameters:

$$P(p_a|c_a, l=1) \qquad P(l=1|k_c)$$

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

M-step

- **Goal**: given $P(l=1|p,c,k)$, estimate model parameters:

$$P(p_a|c_a,l=1) \qquad P(l=1|k_c)$$

$$P(p_{Per}|c_{Per},l=1)$$
$$P(p_{Gen}|c_{Gen},l=1)$$
$$P(p_{Num}|c_{Num},l=1)$$
$$P(p_{Ani}|c_{Ani},l=1)$$

Composed of four groups of parameters, one for each attribute

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

M-step

- **Goal**: given $P(l=1|p,c,k)$, estimate model parameters:

$$P(p_a|c_a,l=1) \qquad P(l=1|k_c)$$

$$P(p_{Per}|c_{Per},l=1)$$
$$P(p_{Gen}|c_{Gen},l=1)$$
$$P(p_{Num}|c_{Num},l=1)$$
$$P(p_{Ani}|c_{Ani},l=1)$$

Composed of four groups of parameters, one for each attribute

use maximum likelihood estimation

What have we done so far?

- described how EM can be used to learn the parameters of our model for overt pronoun resolution in an unsupervised way

What have we done so far?

- described how EM can be used to learn the parameters of our model for overt pronoun resolution in an unsupervised way

How can we apply this overt pronoun resolution model to resolve AZPs?

Applying the model to resolve AZPs

- Given an AZP z,
  - exhaustively search for the candidate antecedent c and overt pronoun p that maximize $P(l=1 \mid p,c,k)$ when p is used to fill the gap left behind by z

Applying the model to resolve AZPs

- Given an AZP z,
  - exhaustively search for the candidate antecedent c and overt pronoun p that maximize $P(l=1 \mid p,c,k)$ when p is used to fill the gap left behind by z

- since the model is trained on overt pronouns but is applied to ZPs, we have to fill each ZP's gap with every overt pronoun when applying the model

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

What remains to be done?

- Recall that the model parameters are:

$$P(p_a|c_a, l=1) \qquad P(l=1|k_c)$$

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

What remains to be done?

- Recall that the model parameters are:

$$P(p_a | c_a, l=1) \qquad P(l=1 | k_c)$$

$$P(p_{Per} | c_{Per}, l=1)$$
$$P(p_{Gen} | c_{Gen}, l=1)$$
$$P(p_{Num} | c_{Num}, l=1)$$
$$P(p_{Ani} | c_{Ani}, l=1)$$

What remains to be done?

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Recall that the model parameters are:

$$P(p_a | c_a, l = 1) \qquad P(l = 1 | k_c)$$

$$P(p_{Per} | c_{Per}, l = 1)$$
$$P(p_{Gen} | c_{Gen}, l = 1)$$
$$P(p_{Num} | c_{Num}, l = 1)$$
$$P(p_{Ani} | c_{Ani}, l = 1)$$

need to compute the person, gender, number, and animacy of a NP

What remains to be done?

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Recall that the model parameters are:

$$P(p_a | c_a, l=1) \qquad P(l=1 | k_c)$$

$$P(p_{Per} | c_{Per}, l=1)$$
$$P(p_{Gen} | c_{Gen}, l=1)$$
$$P(p_{Num} | c_{Num}, l=1)$$
$$P(p_{Ani} | c_{Ani}, l=1)$$

need to compute the person, gender, number, and animacy of a NP

heuristically or via unsupervised learning

What remains to be done?

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Recall that the model parameters are:

$$P(p_a | c_a, l=1) \qquad P(l=1 | k_c)$$

$$P(p_{Per} | c_{Per}, l=1)$$
$$P(p_{Gen} | c_{Gen}, l=1)$$
$$P(p_{Num} | c_{Num}, l=1)$$
$$P(p_{Ani} | c_{Ani}, l=1)$$

need to compute the person, gender, number, and animacy of a NP

heuristically or via unsupervised learning

What remains to be done?

p: overt pronoun
c: candidate antecedent
k: context
l: whether p&c are coref

- Recall that the model parameters are:

$$P(p_a | c_a, l = 1)$$

$$P(l = 1 | k_c)$$

$$P(p_{Per} | c_{Per}, l = 1)$$
$$P(p_{Gen} | c_{Gen}, l = 1)$$
$$P(p_{Num} | c_{Num}, l = 1)$$
$$P(p_{Ani} | c_{Ani}, l = 1)$$

Probability that p and c are coreferent given their context kc

need to compute the person, gender, number, and animacy of a NP

heuristically or via unsupervised learning

p: overt pronoun

c: candidate antecedent

k: context

l: whether p&c are coref

What remains to be done?

- Recall that the model parameters are:

$$P(p_a | c_a, l = 1)$$

$$P(l = 1 | k_c)$$

$$P(p_{Per} | c_{Per}, l = 1)$$
$$P(p_{Gen} | c_{Gen}, l = 1)$$
$$P(p_{Num} | c_{Num}, l = 1)$$
$$P(p_{Ani} | c_{Ani}, l = 1)$$

How to represent k_c?

need to compute the person, gender, number, and animacy of a NP

heuristically or via unsupervised learning

Representing Context

- 8 features; motivated by previous work on AZP resolution

Distance (1)	Sentence distance between p and c
Positional (1)	whether p is the first word of a sentence; if not, whether p is the first word of an IP
Syntactic (4)	whether the node spanning c has an ancestor NP node; if so, whether this node is a descendant of c's lowest ancestor IP node, …
Grammatical (1)	whether c is a subject whose governing verb is lexically identical to the verb governing p
Semantic (1)	whether c is the closest candidate with subject grammatical role and is semantically compatible with p's governing verb

Plan for the Talk

- Intro to Chinese overt pronouns
- Generative model for overt pronoun resolution
  - Training and application
- Evaluation

Evaluation

- **Goal**: evaluate our unsupervised model

Experimental Setup

- Corpus
  - Chinese portion of the OntoNotes 5.0 corpus
  - Unsupervised training of the overt pronoun resolution model
    - Chinese training set used in the CoNLL 2012 shared task
    - 1,391 documents (13,418 overt pronouns)
  - Testing (Applying the model to resolve AZPs)
    - Chinese development set used in the CoNLL 2012 shared task
    - 172 documents (1,713 AZPs)
- Evaluation measures
  - recall (R), precision (P), and F-measure (F) on resolving AZPs

Three Evaluation Settings

- Setting 1: gold parse trees, gold AZPs
- Setting 2: gold parse trees, system AZPs
- Setting 3: system parse trees, system AZPs

Three Evaluation Settings

- Setting 1: gold parse trees, gold AZPs
- Setting 2: gold parse trees, system AZPs
- Setting 3: system parse trees, system AZPs

Seven Baselines

- 4 simple heuristic baselines
  - gauge the difficulty of the resolution task
- 3 state-of-the-art supervised Chinese AZP resolvers

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**

- Baseline 2 performs significantly better than Baseline 1
  - salience plays a greater role than recency

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**
Selecting closest sem. compat. cand.	28.5	28.8	**28.7**	11.7	7.6	**9.2**

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**
Selecting closest sem. compat. cand.	28.5	28.8	**28.7**	11.7	7.6	**9.2**
Selecting closest sem. compat. subject	45.2	45.7	**45.5**	18.9	12.3	**14.9**

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**
Selecting closest sem. compat. cand.	28.5	28.8	**28.7**	11.7	7.6	**9.2**
Selecting closest sem. compat. subject	45.2	45.7	**45.5**	18.9	12.3	**14.9**

- The last two heuristic baselines are created by adding semantic compatibility to the first two heuristic baselines
  - The last two baselines outperform the first two baselines
  - Semantic compatibility is useful for Chinese AZP resolution

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**
Selecting closest sem. compat. cand.	28.5	28.8	**28.7**	11.7	7.6	**9.2**
Selecting closest sem. compat. subject	45.2	45.7	**45.5**	18.9	12.3	**14.9**

- Best heuristic baseline (Baseline 4) uses both salience and semantic compatibility

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**
Selecting closest sem. compat. cand.	28.5	28.8	**28.7**	11.7	7.6	**9.2**
Selecting closest sem. compat. subject	45.2	45.7	**45.5**	18.9	12.3	**14.9**
Duplicated Zhao and Ng (2007)	41.5	41.5	**41.5**	12.7	14.2	**13.4**

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**
Selecting closest sem. compat. cand.	28.5	28.8	**28.7**	11.7	7.6	**9.2**
Selecting closest sem. compat. subject	45.2	45.7	**45.5**	18.9	12.3	**14.9**
Duplicated Zhao and Ng (2007)	41.5	41.5	**41.5**	12.7	14.2	**13.4**

- proposed first learning-based approach to Chinese AZP resolution
- performs worse than the best heuristic baseline

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**
Selecting closest sem. compat. cand.	28.5	28.8	**28.7**	11.7	7.6	**9.2**
Selecting closest sem. compat. subject	45.2	45.7	**45.5**	18.9	12.3	**14.9**
Duplicated Zhao and Ng (2007)	41.5	41.5	**41.5**	12.7	14.2	**13.4**
Duplicated Kong and Zhou (2010)	44.9	44.9	**44.9**	18.7	11.9	**14.5**

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**
Selecting closest sem. compat. cand.	28.5	28.8	**28.7**	11.7	7.6	**9.2**
Selecting closest sem. compat. subject	45.2	45.7	**45.5**	18.9	12.3	**14.9**
Duplicated Zhao and Ng (2007)	41.5	41.5	**41.5**	12.7	14.2	**13.4**
Duplicated Kong and Zhou (2010)	44.9	44.9	**44.9**	18.7	11.9	**14.5**

- resolves AZPs using parse trees as structured features
- Performs better than Zhao and Ng (2007)
- still underperforms best heuristic baseline

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**
Selecting closest sem. compat. cand.	28.5	28.8	**28.7**	11.7	7.6	**9.2**
Selecting closest sem. compat. subject	45.2	45.7	**45.5**	18.9	12.3	**14.9**
Duplicated Zhao and Ng (2007)	41.5	41.5	**41.5**	12.7	14.2	**13.4**
Duplicated Kong and Zhou (2010)	44.9	44.9	**44.9**	18.7	11.9	**14.5**
Chen and Ng (2013)	47.7	47.7	**47.7**	14.9	16.7	**15.7**

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**
Selecting closest sem. compat. cand.	28.5	28.8	**28.7**	11.7	7.6	**9.2**
Selecting closest sem. compat. subject	45.2	45.7	**45.5**	18.9	12.3	**14.9**
Duplicated Zhao and Ng (2007)	41.5	41.5	**41.5**	12.7	14.2	**13.4**
Duplicated Kong and Zhou (2010)	44.9	44.9	**44.9**	18.7	11.9	**14.5**
Chen and Ng (2013)	47.7	47.7	**47.7**	14.9	16.7	**15.7**

- our supervised resolver
- improves Zhao and Ng's (2007) system with two extensions
- best of the 7 baselines

Baselines: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Selecting closest candidate	25.0	25.2	**25.1**	10.3	6.7	**8.1**
Selecting closest subject	42.0	43.6	**42.8**	18.0	11.9	**14.4**
Selecting closest sem. compat. cand.	28.5	28.8	**28.7**	11.7	7.6	**9.2**
Selecting closest sem. compat. subject	45.2	45.7	**45.5**	18.9	12.3	**14.9**
Duplicated Zhao and Ng (2007)	41.5	41.5	**41.5**	12.7	14.2	**13.4**
Duplicated Kong and Zhou (2010)	44.9	44.9	**44.9**	18.7	11.9	**14.5**
Chen and Ng (2013)	47.7	47.7	**47.7**	14.9	16.7	**15.7**
Our unsupervised model	47.5	47.9	**47.7**	19.9	12.9	**15.7**

- performs as well as the best baseline
  - though with a higher recall and lower precision under Setting 3

Ablation Experiments

- in each experiment, remove exactly one probability term from our model and retrain the model

Ablation Experiments: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Full model	47.5	47.9	**47.7**	19.9	12.9	**15.7**
without Number	47.5	47.9	**47.7**	19.7	12.8	**15.5**
without Gender	44.5	45.0	**44.7**	19.2	12.5	**15.1**
without Person	45.2	45.6	**45.4**	19.1	12.4	**15.1**
without Animacy	45.1	45.5	**45.3**	19.1	12.4	**15.1**
without Context features	32.9	33.1	**33.0**	15.2	9.8	**11.9**

Ablation Experiments: Results

	Gold Parses, Gold AZPs			System Parses, System AZPs		
	R	P	F	R	P	F
Full model	47.5	47.9	**47.7**	19.9	12.9	**15.7**
without Number	47.5	47.9	**47.7**	19.7	12.8	**15.5**
without Gender	44.5	45.0	**44.7**	19.2	12.5	**15.1**
without Person	45.2	45.6	**45.4**	19.1	12.4	**15.1**
without Animacy	45.1	45.5	**45.3**	19.1	12.4	**15.1**
without Context features	32.9	33.1	**33.0**	15.2	9.8	**11.9**

- no drop in performance when Number is ablated
- performance drops significantly when other terms are ablated
  - context features contributed the most to overall performance

Major Sources of Error

- Failure in tracking the discourse entity in focus
- Errors in computing semantic compatibility
- Assumption that overt pronouns and ZPs occur in the same context is not always correct

*pro*不客气。
(*pro* are welcome.)

  - In Chinese, this sentence is never used with an overt pronoun, so the overt pronoun resolver learned from unannotated data will not have the knowledge needed to resolve this ZP

Major Sources of Error

- Failure in tracking the discourse entity in focus
- Errors in computing semantic compatibility
- Assumption that overt pronouns and ZPs occur in the same context is not always correct

  *pro*不客气。
  (*pro* are welcome.)

  - In Chinese, this sentence is never used with an overt pronoun, so the overt pronoun resolver learned from unannotated data will not have the knowledge needed to resolve this ZP
  - This is a case that can be easily handled by the supervised resolvers but not by our model

Summary

- Proposed an unsupervised model for AZP resolution
  - rivaled its supervised counterparts in performance when evaluated on the Chinese portion of the OntoNotes v5.0 corpus