



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

Chen Chen and Vincent Ng

Human Language Technology Research Institute
University of Texas at Dallas

Chinese Common Noun Phrase (NP) Resolution

- Find an **antecedent** for each common NP in Chinese text
 - the preceding noun phrase it refers to

Chinese Common Noun Phrase (NP) Resolution

- Find an **antecedent** for each common NP in Chinese text
 - **the preceding noun phrase it refers to**

蓝欣无线全年亏损 1 4 0 0 万元。
企业接近破产的边缘。

Lanxin Wireless lost 14 million dollars in one year.
The company is on the verge of bankruptcy.

Chinese Common Noun Phrase (NP) Resolution

- Find an **antecedent** for each common NP in Chinese text
 - **the preceding noun phrase it refers to**

蓝欣无线全年亏损 1 4 0 0 万元。
企业接近破产的边缘。

Lanxin Wireless lost 14 million dollars in one year.
The company is on the verge of bankruptcy.

Chinese Common Noun Phrase (NP) Resolution

- Find an **antecedent** for each common NP in Chinese text
 - **the preceding noun phrase it refers to**

蓝欣无线全年亏损 1 4 0 0 万元。
企业接近破产的边缘。

Lanxin Wireless lost 14 million dollars in one year.
The company is on **the verge of bankruptcy**.

Chinese Common NP Resolution: Two Subtasks

- Given a Chinese common NP n ,
 1. determine whether n has an antecedent
 2. if yes, determine which preceding NP is its antecedent

Chinese Common NP Resolution: Two Subtasks

- Given a Chinese common NP n ,
 1. determine whether n has an antecedent
 2. if yes, determine which preceding NP is its antecedent
- Most previous work addresses them in a **pipeline** fashion
 - **Weakness**: error propagates from 1st subtask to 2nd subtask
 - We perform the two subtasks in a **joint** fashion

Common NP Resolution is Challenging

- Requires **lexical semantic** and **world** knowledge
 - **Lanxin Wireless** vs. the company ✓
 - **money bank** vs. river bank ✗

Common NP Resolution is Challenging

- Requires **lexical semantic** and **world** knowledge
 - **Lanxin Wireless** vs. the company ✓
 - **money bank** vs. **river bank** ✗
- Common NP resolution in **Chinese** is even more challenging
 - scarcity of Chinese lexical knowledge bases for providing world knowledge

Common NP Resolution is Challenging

- Requires **lexical semantic** and **world** knowledge
 - **Lanxin Wireless** vs. **the company** ✓
 - **money bank** vs. **river bank** ✗
- Common NP resolution in **Chinese** is even more challenging
 - scarcity of Chinese lexical knowledge bases for providing world knowledge
 - This problem can be mitigated in part by using annotated data
 - data where each common NP is annotated with its antecedent

Common NP Resolution is Challenging

- Requires **lexical semantic** and **world** knowledge
 - **Lanxin Wireless** vs. **the company** ✓
 - **money bank** vs. **river bank** ✗
- Common NP resolution in **Chinese** is even more challenging
 - scarcity of Chinese lexical knowledge bases for providing world knowledge
 - This problem can be mitigated in part by using annotated data
 - data where each common NP is annotated with its antecedent
 - But.. we made it challenging by **not using annotated data**
 - **Unsupervised common NP resolution**

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:
 1. Train a **pairwise model** to determine the probability that a common NP n and a candidate antecedent c given their context k are coreferent, i.e., $P(\text{coref}=+|n,c,k)$

If we had annotated training data ...

- we could adopt the standard **supervised** approach:
 1. Train a **pairwise model** to determine the probability that a common NP n and a candidate antecedent c given their context k are coreferent, i.e., $P(\text{coref}=+|n,c,k)$

[Lanxin Wireless] lost [14 million dollars] in one year.
[The company] is on [the verge of bankruptcy].

Training Instances:

coref?	Common NP	Candidate Antecedent
+	The company	Lanxin Wireless
-	The company	14 million dollars
-	the verge of bankruptcy	Lanxin Wireless
-	the verge of bankruptcy	14 million dollars
-	the verge of bankruptcy	the company

If we had annotated training data ...

- we could adopt the standard **supervised** approach:
 1. Train a **pairwise model** to determine the probability that a common NP n and a candidate antecedent c given their context k are coreferent, i.e., $P(\text{coref}=+|n,c,k)$
 2. Apply the model to each common NP to select the candidate with the highest probability as its antecedent

But we don't have annotated data...

coref?	Common NP	Candidate Antecedent
?	The company	Lanxin Wireless
?	The company	14 million dollars
?	the verge of bankruptcy	Lanxin Wireless
?	the verge of bankruptcy	14 million dollars
?	the verge of bankruptcy	the company

But we don't have annotated data...

coref?	Common NP	Candidate Antecedent
?	The company	Lanxin Wireless
?	The company	14 million dollars
?	the verge of bankruptcy	Lanxin Wireless
?	the verge of bankruptcy	14 million dollars
?	the verge of bankruptcy	the company

- **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 NPs is coreferent

But we don't have annotated data...

coref?	Common NP	Candidate Antecedent
0.85	The company	Lanxin Wireless
0.15	The company	14 million dollars
0.01	the verge of bankruptcy	Lanxin Wireless
0.05	the verge of bankruptcy	14 million dollars
0.08	the verge of bankruptcy	the company

- **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 NPs is coreferent

But we don't have annotated data...

coref?	Common NP	Candidate Antecedent
0.85	The company	Lanxin Wireless
0.15	The company	14 million dollars
0.01	the verge of bankruptcy	Lanxin Wireless
0.05	the verge of bankruptcy	14 million dollars
0.08	the verge of bankruptcy	the company

- **Idea:** design a **generative** model and use **EM** to **iteratively**
 - Fill in missing values probabilistically (**E-step**)
 - Estimate model parameters using the filled values (**M-step**)

But we don't have annotated data...

coref?	Common NP	Candidate Antecedent
0.85	The company	Lanxin Wireless
0.15	The company	14 million dollars
0.01	the verge of bankruptcy	Lanxin Wireless
0.05	the verge of bankruptcy	14 million dollars
0.08	the verge of bankruptcy	the company

- Recall that we **jointly** perform two subtasks
 - Determine whether a common NP has an antecedent
 - Find the antecedent

But we don't have annotated data...

coref?	Common NP	Candidate Antecedent
0.85	The company	Lanxin Wireless
0.15	The company	14 million dollars
0.01	the verge of bankruptcy	Lanxin Wireless
0.05	the verge of bankruptcy	14 million dollars
0.08	the verge of bankruptcy	the company

- How to perform them **jointly**?
 - Introduce a *dummy* candidate antecedent for each common NP

But we don't have annotated data...

coref?	Common NP	Candidate Antecedent
0.85	The company	Lanxin Wireless
0.15	The company	14 million dollars
0.01	the verge of bankruptcy	Lanxin Wireless
0.05	the verge of bankruptcy	14 million dollars
0.08	the verge of bankruptcy	the company
0.01	The company	dummy
0.86	the verge of bankruptcy	dummy

- How to perform them **jointly**?
 - Introduce a *dummy* candidate antecedent for each common NP

But we don't have annotated data...

coref?	Common NP	Candidate Antecedent
0.85	The company	Lanxin Wireless
0.15	The company	14 million dollars
0.01	the verge of bankruptcy	Lanxin Wireless
0.05	the verge of bankruptcy	14 million dollars
0.08	the verge of bankruptcy	the company
0.01	The company	dummy
0.86	the verge of bankruptcy	dummy

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

Plan for the talk

- Generative model
 - EM training
- Evaluation

Plan for the talk

- Generative model
 - EM training
- Evaluation

Goal

- fill in the missing class values probabilistically
 - i.e., compute $P(\text{coref}=+|n,c,k)$
 - n: common NP
 - c: candidate antecedent
 - k: context

Goal

- fill in the missing class values probabilistically
 - i.e., compute $P(\text{coref}=+|n,c,k)$
 - n: common NP
 - c: candidate antecedent
 - k: context
- Using Chain Rule,

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

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

Goal

- fill in the missing class values probabilistically
 - i.e., compute $P(\text{coref}=+|n,c,k)$
 - n: common NP
 - c: candidate antecedent
 - k: context
- Using Chain Rule,

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

- Applying Chain Rule to the numerator,

$$\begin{aligned} & P(n, c, k, \text{coref} = +) \\ &= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k) \end{aligned}$$

Goal

- fill in the missing class values probabilistically
 - i.e., compute $P(\text{coref}=+|n,c,k)$
 - n: common NP
 - c: candidate antecedent
 - k: context
- Using Chain Rule,

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

- Applying Chain Rule to the numerator,

$$\begin{aligned} &P(n, c, k, \text{coref} = +) \\ &= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k) \end{aligned}$$

This is our generative model!

Generative Model

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$

Generative Model

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



generate
context k

Generative Model

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



generate
candidate c
given context k

Generative Model

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



generate class label
given candidate c
and context k

Generative Model

n: common NP
c: candidate antecedent
k: context

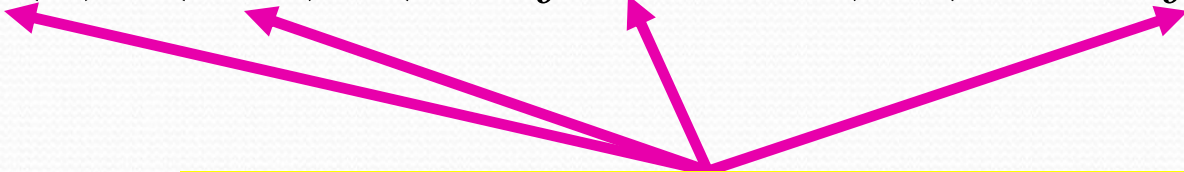
$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



generate common NP n
given class label,
candidate c and
context k

Generative Model


n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$


These four are the **model parameters**

How to **estimate** each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$


These four are the **model parameters**

How to estimate each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Assumption: for each common NP, the contexts generated from different candidate antecedents have the same probability

- Effectively ignoring this term

How to **estimate** each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +) \\ = P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Prior probability of a candidate antecedent c given context k

- if c is an **implausible** candidate antecedent, we set $P(c|k)$ to 0
- otherwise, we set $P(c|k)$ to 1

How to **estimate** each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +) \\ = P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Prior probability of a candidate antecedent c given context k

- if c is an **implausible** candidate antecedent, we set $P(c|k)$ to 0
- otherwise, we set $P(c|k)$ to 1

How to identify implausible candidate antecedents?

- Use **linguistic constraints** on coreference
 - e.g., compatibility w.r.t. gender, number, semantic class

How to **estimate** each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Prior probability that they are coreferent given candidate & context

How to **estimate** each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Prior probability that they are coreferent given candidate & context

How to estimate this probability?

How to **estimate** each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +) \\ = P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Prior probability that they are coreferent given candidate & context

How to estimate this probability?

- represent context k using 5 commonly-used features
 - sentence distance between c and n
 - whether the governing verbs of c and n are the same
 - ...

How to **estimate** each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +) \\ = P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Prior probability that they are coreferent given candidate & context

How to estimate this probability?

- represent context k using 5 commonly-used features
 - sentence distance between c and n
 - whether the governing verbs of c and n are the same
 - ...
- estimate probability in the **M-step**

How to **estimate** each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Probability of a common NP given everything else

How to estimate each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Probability of a common NP given everything else

How to estimate $P(n | \text{coref} = +, c, k)$?

How to estimate each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Probability of a common NP given everything else

How to estimate $P(n | \text{coref} = +, c, k)$?

- simplify by dropping k, yielding

$$P(n | \text{coref} = +, c)$$

How to estimate each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +) \\ = P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Probability of a common NP given everything else

How to estimate $P(n | \text{coref} = +, c, k)$?

- simplify by dropping k, yielding

$$P(n | \text{coref} = +, c)$$

- approximate n and c by their head nouns, yielding

$$P(n_h | \text{coref} = +, c_h)$$

How to estimate each of these parameters?

n: common NP
c: candidate antecedent
k: context

$$P(n, c, k, \text{coref} = +)$$
$$= P(k)P(c | k)P(\text{coref} = + | c, k)P(n | \text{coref} = +, c, k)$$



Probability of a common NP given everything else

How to estimate $P(n | \text{coref} = +, c, k)$?

- simplify by dropping k, yielding

$$P(n | \text{coref} = +, c)$$

- approximate n and c by their head nouns, yielding

$$P(n_h | \text{coref} = +, c_h)$$

- estimate $P(n_h | \text{coref} = +, c_h)$ in the **M-step**

The EM Algorithm

- E-step:
 - Fill in the missing class values probabilistically by computing $P(\text{coref}=+|n,c,k)$ using the current model parameter values
- M-step:
 - Re-estimate the model parameters using **maximum likelihood estimation**

Applying the Learned Model to Test Data

- Use the model to compute the probability that each common NP n is coreferent with each candidate antecedent c
- For each n , pick c with highest probability as its antecedent

Plan for the talk

- Generative model
 - EM training
- Evaluation

Evaluation

- **Goal:** evaluate our unsupervised model

Evaluation Setup

- **Corpus**
 - Chinese portion of the OntoNotes 5.0 corpus
 - **Unsupervised training** of the common NP resolution model
 - Chinese training and dev sets used in CoNLL 2012 shared task
 - 1,563 documents
 - **Testing** (Apply the model to resolve common NPs)
 - Chinese test set used in the CoNLL 2012 shared task
 - 166 documents
- **Evaluation measures**
 - Recall (R), precision (P), and F-measure (F) on resolving common NPs

Baseline Systems

- Heuristic baseline
 - resolve to closest preceding NP with the same head
- State-of-the-art supervised resolver
 - Björkelund and Kuhn (2014)

Results

System	R	P	F
Heuristic baseline	49.3	28.9	36.5
Supervised baseline	42.0	50.6	45.9
Our System	46.2	47.5	46.8

Summary

- Proposed an unsupervised model for Chinese common NP resolution
 - rivaled best existing supervised resolver in performance when evaluated on the Chinese portion of the OntoNotes 5.0 corpus