Relation Extraction

What is relation extraction?

Dan Jurafsky

Relations Extracting:
從一段非結構化的文章當中，找出他的主題，並轉換為結構化的資料
最後從這些結構化的資料找出Name Entities，並分析他們的relation

Extracting relations from text

- Company report: "International Business Machines Corporation (IBM or the company) was incorporated in the State of New York on June 16, 1911, as the Computing-Tabulating-Recording Co. (C-T-R)…"
- Extracted Complex Relation:

  Company-Founding

Company	IBM
Location	New York
Date	June 16, 1911
Original-Name	Computing-Tabulating-Recording Co.

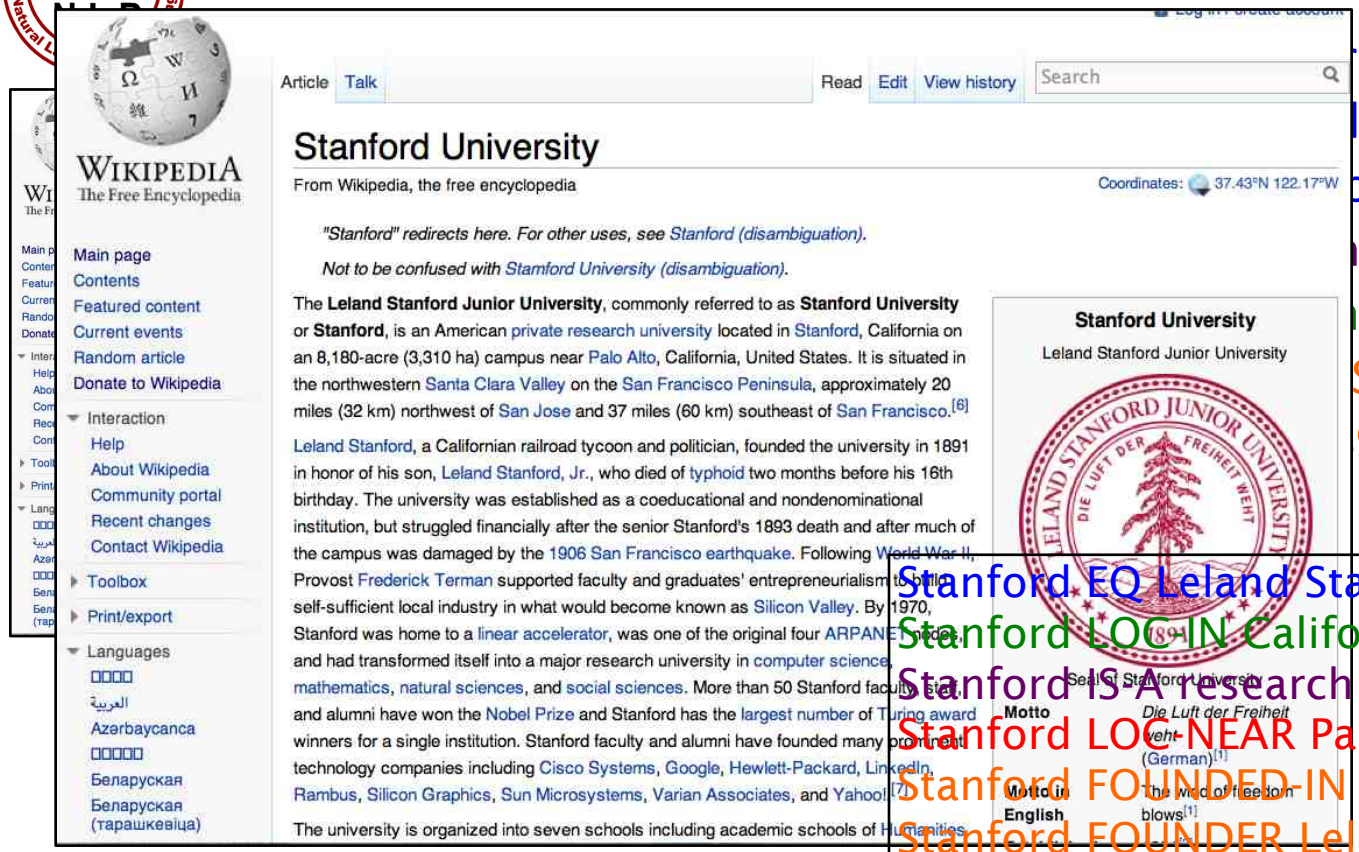
- But we will focus on the simpler task of extracting relation **triples**

  Founding-year(IBM,1911)

  Founding-location(IBM,New York)

e.g. 一篇講解IBM的文章，可以從中萃取出IBM的名字、創辦地點、創辦日、創辦人、原始名稱等等

Extracting Relation Triples from Text



d Junior University,
l to as Stanford
ord, is an American
iversity located in
… near Palo Alto,
Stanford…founded
91

Stanford EQ Leland Stanford Junior University
Stanford LOC-IN California
Stanford IS-A research university
Stanford LOC-NEAR Palo Alto
Stanford FOUNDED-IN 1891
Stanford FOUNDER Leland Stanford

一篇講解Stanford的文章
可以從中萃取出他的地點(located in …)
全名、創辦日、創辦人...

把萃取出來的資料轉為結構化的資料，存入資料庫中，即可進行分析
RE可以用來設計問答機器人

Dan Jurafsky

Why Relation Extraction?

給電腦讀的WikiPedia
以結構化資料為主

- Create new structured knowledge bases, useful for any app
- Augment current knowledge bases
  - Adding words to WordNet thesaurus, facts to FreeBase or DBPedia
- Support question answering
  - The granddaughter of which actor starred in the movie "E.T."?

```
(acted-in ?x "E.T.")(is-a ?y actor)(granddaughter-of ?x ?y)
```

- But which relations should we extract?

?x acted–in "E.T."
?y is–a actor
?x is granddaughter–of ?y

對QA系統，針對問的問題，從資料庫找出符合的pattern與資料，即可找出缺少的變數

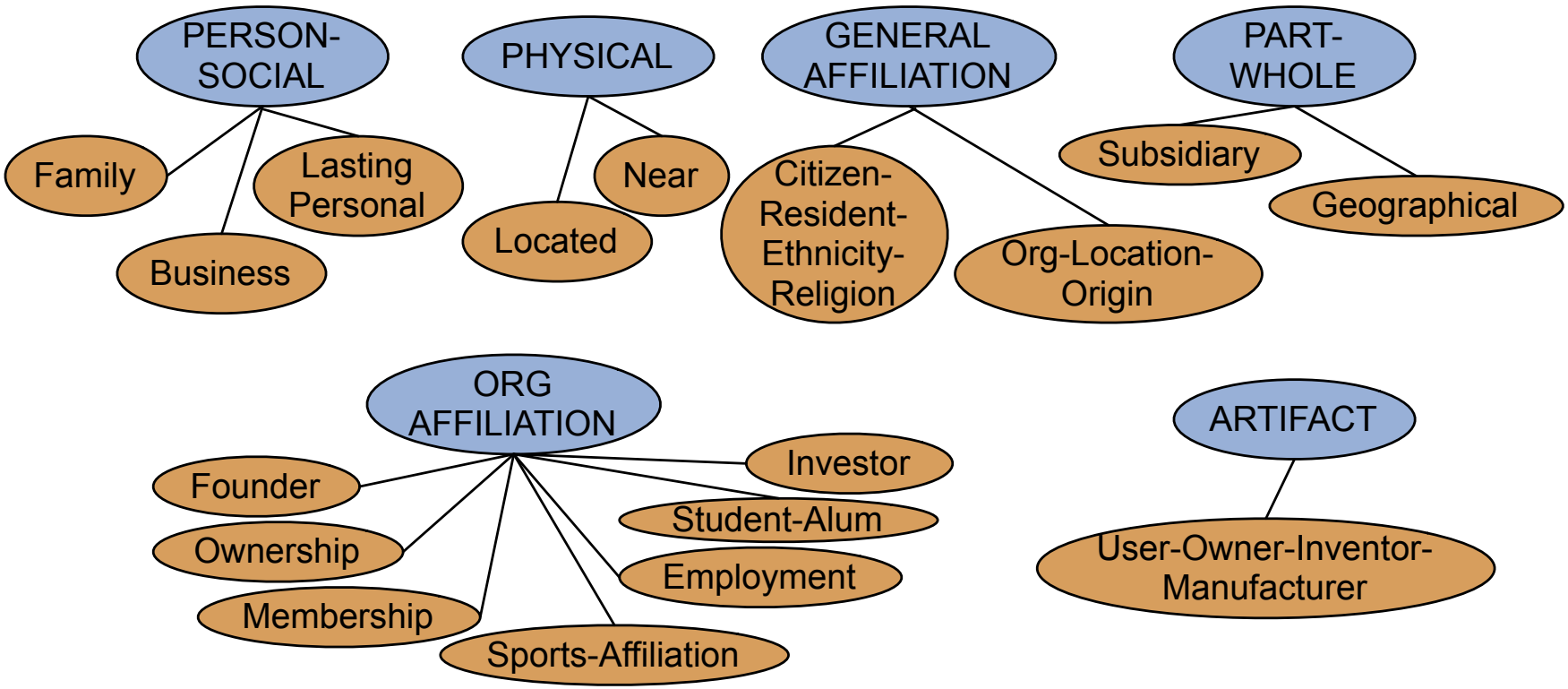
Dan Jurafsky

RE常用來萃取出來的類別與其關係

Automated Content Extraction (ACE)

17 relations from 2008 "Relation Extraction Task"

針對不同的領域，會有不同的類別關係



Automated Content Extraction (ACE)

- Physical-Located PER-GPE
  He was in Tennessee
  人物–所在地點
- Part-Whole-Subsidiary ORG-ORG
  XYZ, the parent company of ABC
  子公司–母公司
- Person-Social-Family PER-PER
  John's wife Yoko
  家庭成員關係
- Org-AFF-Founder PER-ORG
  Steve Jobs, co-founder of Apple…
  公司–創辦人
-

Dan Jurafsky

UMLS: Unified Medical Language System

主要用於藥學、醫學

- 134 entity types, 54 relations

Injury	disrupts	Physiological Function
Bodily Location	location-of	Biologic Function
Anatomical Structure	part-of	Organism
Pharmacologic Substance	causes	Pathological Function
Pharmacologic Substance	treats	Pathologic Function

受傷 破壞 身體機能
身體部位 用於 生物機能
解剖結構 是一部份的 有機體
藥物 引發 病理功能
藥物 治療 病理功能

Extracting UMLS relations from a sentence

Doppler echocardiography can be used to diagnose left anterior descending artery stenosis in patients with type 2 diabetes

↓

Echocardiography, Doppler DIAGNOSES Acquired stenosis

Dan Jurafsky

Databases of Wikipedia Relations

Wikipedia Infobox

```
{{Infobox university
|image_name= Stanford University seal.svg
|image_size= 210px
|caption = Seal of Stanford University
|name =Stanford University
|native_name =Leland Stanford Junior Uni
|motto = {{lang|de|"Die Luft der Freiheit v
name="casper">{{cite speech|title=Die Lu
Casper|first=Gerhard|last=Casper|author
05|url=http://www.stanford.edu/dept/pr
|mottoeng = The wind of freedom blows<
|established = 1891<ref>{{cite web |
url=http://www.stanford.edu/home/stan
publisher = Stanford University | accessda
|type = [[private university|Private]]
|calendar= Quarter
|president = [[John L. Hennessy]]
|provost = [[John Etchemendy]]
|city = [[Stanford, California|Stanford]]
|state = California
|country = U.S.
```

Type	Private
Endowment	US$ 16.5 billion (2011)[3]
President	John L. Hennessy
Provost	John Etchemendy
Academic staff	1,910[4]
Students	15,319
Undergraduates	6,878[5]
Postgraduates	8,441[5]
Location	Stanford, California, U.S.
Campus	Suburban, 8,180 acres (3,310 ha)[6]
Colors	Cardinal red and white

Relations extracted from Infobox
Stanford state California
Stanford motto "Die Luft der Freiheit weht"

wiki結構化的資料

Dan Jurafsky

Relation databases that draw from Wikipedia

- Resource Description Framework (RDF) triples

  relation

  subject predicate object

  Golden Gate Park和San Francisco有location關係
  之後就把這個關係關換成dbpedia格式並存起來

  `Golden Gate Park location San Francisco`

  dbpedia:Golden_Gate_Park dbpedia-owl:location dbpedia:San_Francisco

- DBPedia: 1 billion RDF triples, 385 from English Wikipedia

  millions

- Frequent Freebase relations:

  people/person/nationality,
  people/person/profession,
  biology/organism_higher_classification

  location/location/contains
  people/person/place-of-birth
  film/film/genre

Ontological relations

Examples from the WordNet Thesaurus

- IS-A (hypernym): subsumption between classes
  - `Giraffe` IS-A `ruminant` IS-A `ungulate` IS-A `mammal` IS-A `vertebrate` IS-A `animal`…
    - 小類別 is a 大類別 V
    - 大類別 is a 小類別 X
- Instance-of: relation between individual and class
  - `San Francisco` instance-of `city`
    - 物件 is instance of 類別
    - 小類別 is instance of 大類別

How to build relation extractors

1. Hand-written patterns
2. Supervised machine learning
3. Semi-supervised and unsupervised
   - Bootstrapping (using seeds)
   - Distant supervision
   - Unsupervised learning from the web

Relation Extraction

What is relation extraction?

Relation Extraction

Using patterns to extract relations

Rules for extracting IS-A relation

Hand-written pattern

Early intuition from **Hearst (1992)**

- "Agar is a substance prepared from a mixture of red algae, such as Gelidium, for laboratory or industrial use"

- What does *Gelidium* mean?
- How do you know?`

不知道Gelidium的意思，但從文章可以發現，Gelidium就是一種red algae
Gelidium is a red algae
因為有such as做連接

Rules for extracting IS-A relation

Early intuition from **Hearst (1992)**

- "Agar is a substance prepared from a mixture of **red algae, such as Gelidium,** for laboratory or industrial use"

- What does *Gelidium* mean?
- How do you know?`

Dan Jurafsky

Hearst's Patterns for extracting IS-A relations

(Hearst, 1992): Automatic Acquisition of Hyponyms

X is a Y

"Y such as X ((, X)* (, and|or) X)"
"such Y as X"
"X or other Y"
"X and other Y"
"Y including X"
"Y, especially X"

Hearst's Patterns for extracting IS-A relations

Hearst pattern	Example occurrences
X and other Y	...temples, treasuries, and other important civic buildings.
X or other Y	Bruises, wounds, broken bones or other injuries...
Y such as X	The bow lute, such as the Bambara ndang...
Such Y as X	...such authors as Herrick, Goldsmith, and Shakespeare.
Y including X	...common-law countries, including Canada and England...
Y , especially X	European countries, especially France, England, and Spain...

Extracting Richer Relations Using Rules

- Intuition: relations often hold between specific entities
  - located-in (ORGANIZATION, LOCATION)
  - founded (PERSON, ORGANIZATION)
  - cures (DRUG, DISEASE)
- Start with Named Entity tags to help extract relation!

Dan Jurafsky

Named Entities aren't quite enough.
Which relations hold between 2 entities?

光用Name Entities不夠
兩個Name Entities可能會有多種關係
要確認究竟是哪一種關係？

Drug

Cure?

Prevent?

Cause?

Disease

Dan Jurafsky

What relations hold between 2 entities?

PERSON

Founder?

Investor?

Member?

Employee?

President?

ORGANIZATION

光用Name Entities不夠
兩個Name Entities可能會有多種關係
要確認究竟是哪一種關係？

Dan Jurafsky

Extracting Richer Relations Using Rules and Named Entities

透過前後文來解決關係消歧異的問題

Who holds what office in what organization?

PERSON, POSITION of ORG

- George Marshall, Secretary of State of the United States

PERSON (named|appointed|chose|*etc.*) PERSON Prep? POSITION

- Truman appointed Marshall Secretary of State

PERSON [be]? (named|appointed|*etc.*) Prep? ORG POSITION

- George Marshall was named US Secretary of State

  Name: George Marshall
  ORG: US
  Position: Secretary of State

Hand-built patterns for relations

- Plus:
  - Human patterns tend to be high-precision
  - Can be tailored to specific domains 各領域特別的用法都可適用
- Minus
  - Human patterns are often low-recall
  - A lot of work to think of all possible patterns!
  - Don't want to have to do this for every relation!
  - We'd like better accuracy

只能找出少數特定用字的關係，因為你把連接詞寫死了

Relation Extraction

Using patterns to extract relations

Relation Extraction

Supervised relation extraction

Supervised machine learning for relations

- Choose a set of relations we'd like to extract
- Choose a set of relevant named entities
- Find and label data
  - Choose a representative corpus
  - Label the named entities in the corpus
  - Hand-label the relations between these entities
  - Break into training, development, and test
- Train a classifier on the training set

必須先準備一些標記好的Name entities與他們之間的關係拿來訓練

1. 找出兩個name entities
2. 判斷兩者有沒有relation
3. 若有，那是哪一種relation

Dan Jurafsky

How to do classification in supervised relation extraction

1. Find all pairs of named entities (usually in same sentence)
2. Decide if 2 entities are related
3. If yes, classify the relation

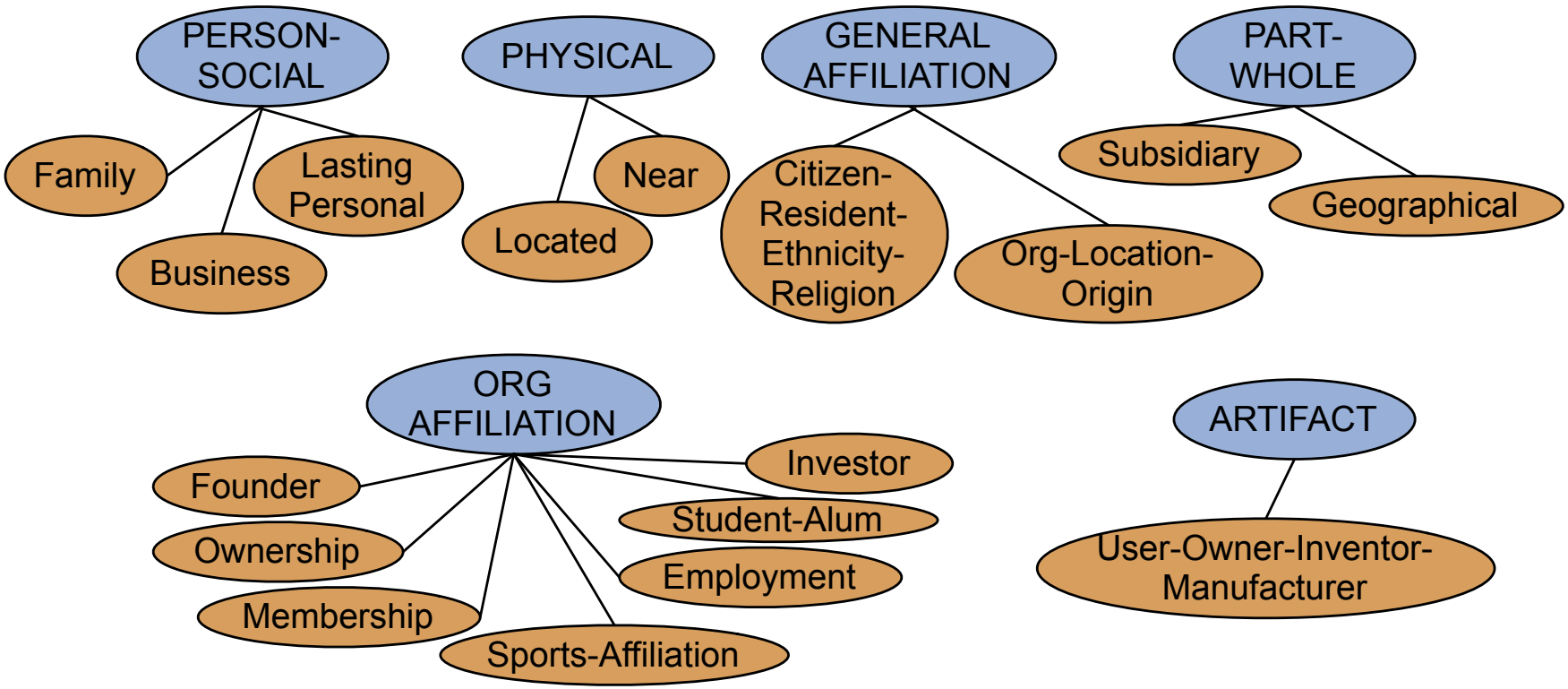
- Why the extra step?
  - Faster classification training by eliminating most pairs
  - Can use distinct feature-sets appropriate for each task.

第一個classifier去判斷是否有關連
另一個classifier去判斷是什麼關連

why 2 classifiers?
多數的named entities都沒有關聯
這樣第一個classifier可以刪除多數組合
讓第二個classifier能快速判斷關聯

Automated Content Extraction (ACE)

17 sub-relations of 6 relations from 2008 "Relation Extraction Task"



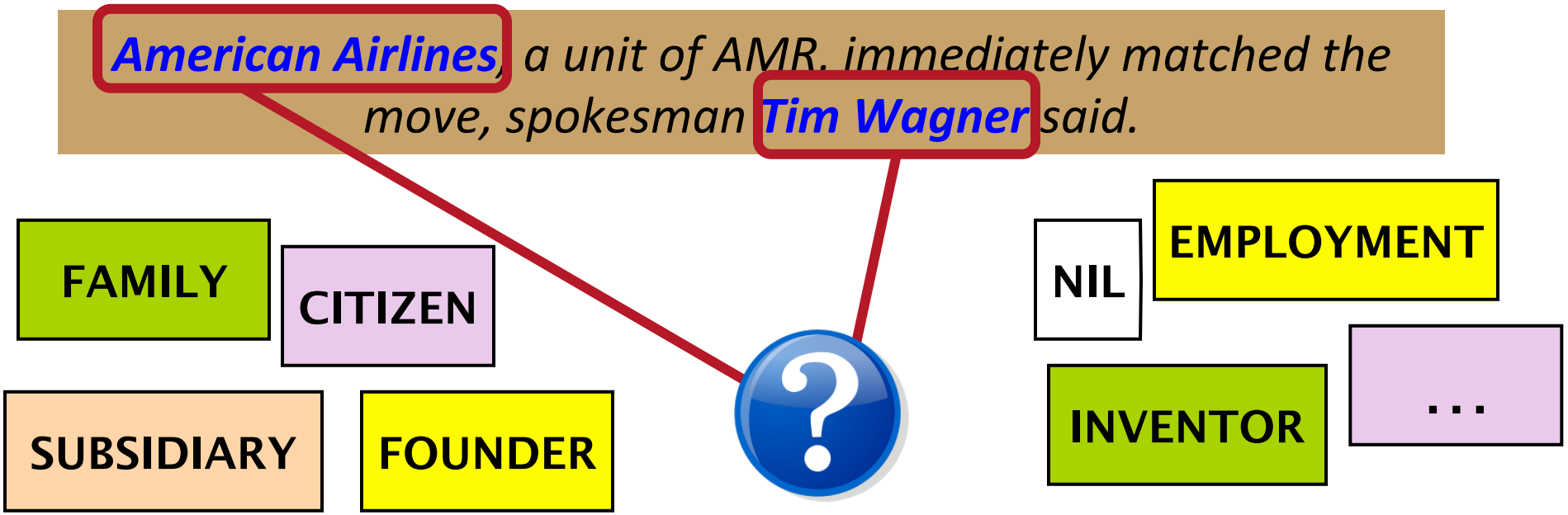
Dan Jurafsky

Relation Extraction

在文中找到了兩個entities，要如何判斷他們的關聯？

Classify the relation between two entities in a sentence

American Airlines, a unit of AMR, immediately matched the move, spokesman Tim Wagner said.



如何分析兩個Entity的關係？
分析文本中兩個Entity中間的文字

Dan Jurafsky

Word Features for Relation Extraction

American Airlines, *a unit of AMR, immediately matched the move, spokesman* ***Tim Wagner*** *said*

Mention 1 Mention 2

- Headwords of M1 and M2, and combination
  Airlines Wagner Airlines-Wagner three features
- Bag of words and bigrams in M1 and M2 M1,M2的排列組合之集合
  {American, Airlines, Tim, Wagner, American Airlines, Tim Wagner}
- Words or bigrams in particular positions left and right of M1/M2
  M2: -1 spokesman
  M2: +1 said
- Bag of words or bigrams between the two entities
  {a, AMR, of, immediately, matched, move, spokesman, the, unit}

先找出headwords的bigram Bag of words
再找出headwords之間的字所構成的Bag of words

Dan Jurafsky

Named Entity Type and Mention Level Features for Relation Extraction

American Airlines, *a unit of AMR, immediately matched the move, spokesman* ***Tim Wagner*** *said*
Mention 1 Mention 2

- Named-entity types
  - M1: ORG American Airline
  - M2: PERSON Tim Wagner

這兩個字的詞性是在一開始的Name Entities Tags就已經成功定義的

- Concatenation of the two named-entity types
  - ORG-PERSON
- Entity Level of M1 and M2 (NAME, NOMINAL, PRONOUN)
  - M1: NAME [it or he would be PRONOUN]
  - M2: NAME [the company would be NOMINAL]

名字 名詞 代詞

先分析指定單字的Name entity，再去分析他們的entity level
最後得知
M1(American Airline) = ORG & NAME
M2(Tim Wagner) = PERSON & NAME

Dan Jurafsky

Parse Features for Relation Extraction

N P N P P P V P N P N P

American Airlines, *a unit of AMR, immediately matched the move, spokesman* ***Tim Wagner*** *said*

Mention 1 Mention 2

- Base syntactic chunk sequence from one to the other

  NP NP PP VP NP NP

  Run a parser to generate a constituent path
- Constituent path through the tree from one to the other

  NP ↑ NP ↑ S ↑ S ↓ NP
- Dependency path

  Airlines matched Wagner said

  N P V P N P V P

Dan Jurafsky

Gazetteer and trigger word features for relation extraction

親屬關係用字，出現這些字的時候就代表某中關係被發現了

- Trigger list for family: kinship terms
  - parent, wife, husband, grandparent, etc. [from WordNet]
- Gazetteer: 地名詞典　　不一定是地名，一串人名也可以稱為Gazetteer
  - Lists of useful geo or geopolitical words
    - Country name list
    - Other sub-entities

把有名的人名、國家名稱、其他專有名詞列出來並存成一個LIST
可以用來更有效的做Name Entity

Trigger list: 某一個領域中常被用到的詞彙，可以用來定義關係的

Dan Jurafsky

***American Airlines**, a unit of AMR, immediately matched the move, spokesman **Tim Wagner** said.*

Entity-based features	
Entity_1 type	ORG
Entity_1 head	*airlines*
Entity_2 type	PERS
Entity_2 head	*Wagner*
Concatenated types	ORGPERS
Word-based features	
Between-entity bag of words	{ *a, unit, of, AMR, Inc., immediately, matched, the, move, spokesman* }
Word(s) before Entity_1	NONE
Word(s) after Entity_2	*said*
Syntactic features	
Constituent path	$NP \uparrow NP \uparrow S \uparrow S \downarrow NP$
Base syntactic chunk path	$NP \rightarrow NP \rightarrow PP \rightarrow NP \rightarrow VP \rightarrow NP \rightarrow NP$
Typed-dependency path	$Airlines \leftarrow_{subj} matched \leftarrow_{comp} said \rightarrow_{subj} Wagner$

現在我們從訓練資料知道
airlines和Wagner有ORG–PERS關聯
所以要設定這個關聯具備的特徵，才能丟進去訓練
特徵可能包含：
兩個Entity各自的前後文
兩個Entity之間的Bag of Words
包含兩個Entity的句子的詞性組成

有了這些features後，就可以訓練出classirier，進而對Name Entity做分類

Classifiers for supervised methods

- Now you can use any classifier you like
  - MaxEnt
  - Naïve Bayes
  - SVM
  - ...
- Train it on the training set, tune on the dev set, test on the test set

Evaluation of Supervised Relation Extraction

- Compute P/R/F_1 for each relation

$$P = \frac{\text{\# of correctly extracted relations}}{\text{Total \# of extracted relations}}$$

$$R = \frac{\text{\# of correctly extracted relations}}{\text{Total \# of gold relations}}$$

$$F_1 = \frac{2PR}{P+R}$$

Summary: Supervised Relation Extraction

if same domain

+ Can get high accuracies with enough hand-labeled training data, if test similar enough to training

- Labeling a large training set is expensive

- Supervised models are brittle, don't generalize well to different genres

如果你丢進去的training data是某一個領域的句子
那對於那一個領域的判斷會很準確
一旦跨到其他領域的話就不一定了

Relation Extraction

Supervised relation extraction

Relation Extraction

Semi-supervised and unsupervised relation extraction

Dan Jurafsky

Seed-based or bootstrapping approaches to relation extraction

- No training set? Maybe you have:
  - A few seed tuples or
  - A few high-precision patterns

seed tuples: 兩個有關係的單字，但並不知道是什麼關係

- Can you use those seeds to do something useful?
  - Bootstrapping: use the seeds to directly learn to populate a relation

只有有限的範例–>用他去學習

用有限的data
丟進DBPedia, FreeBase等
去找出他們的pattern
再透過這些pattern去找出更多的Name Entity pair

Dan Jurafsky

Relation Bootstrapping (Hearst 1992)

- Gather a set of seed pairs that have relation R
- Iterate:
  1. Find sentences with these pairs
  2. Look at the context between or around the pair and generalize the context to create patterns
  3. Use the patterns for grep for more pairs

找出Name Entity的pair
用這些pair的上下文去建立pattern
再用這些pattern去尋找更多pair

Dan Jurafsky

Bootstrapping

- <Mark Twain, Elmira> Seed tuple
  - Grep (google) for the environments of the seed tuple

  "Mark Twain is buried in Elmira, NY."

  X is buried in Y

  "The grave of Mark Twain is in Elmira"

  The grave of X is in Y

  "Elmira is Mark Twain's final resting place"

  Y is X's final resting place.
- Use those patterns to grep for new tuples
- Iterate

我們知道Mark Twain, Elmira有埋葬地關係
那就把這個關係拿去DBPedia, FreeBase
並從回傳的結果可以學習到
__ is buried in __
The grave of __ is in __
__ is __'s final resting place
這三句結構(pattern)代表的意思就是埋葬

接下來就可以再用這三個pattern
去找出其他tuples
(e.g. 其他人被埋葬在什麼地方...)

Dan Jurafsky

Dipre: Extract <author,book> pairs

Brin, Sergei. 1998. Extracting Patterns and Relations from the World Wide Web.

- Start with 5 seeds:

Author	Book
Isaac Asimov	The Robots of Dawn
David Brin	Startide Rising
James Gleick	Chaos: Making a New Science
Charles Dickens	Great Expectations
William Shakespeare	The Comedy of Errors

- Find Instances:

The Comedy of Errors, by William Shakespeare, was
The Comedy of Errors, by William Shakespeare, is
The Comedy of Errors, one of William Shakespeare's earliest attempts
The Comedy of Errors, one of William Shakespeare's most

藉由句子找到pattern
再透過pattern去找更多句子

- Extract patterns (group by middle, take longest common prefix/suffix)

`?x , by ?y ,` `?x , one of ?y 's`

- Now iterate, finding new seeds that match the pattern

再把這兩個patterns丟google，看能不能找出其它name entities

Dan Jurafsky

Snowball

E. Agichtein and L. Gravano 2000. Snowball: Extracting Relations from Large Plain-Text Collections. ICDL

- Similar iterative algorithm

Organization	Location of Headquarters
Microsoft	Redmond
Exxon	Irving
IBM	Armonk

- Group instances w/similar prefix, middle, suffix, extract patterns
  - But require that X and Y be named entities
  - And compute a confidence for each pattern

.69 ORGANIZATION `{'s, in, headquarters}` LOCATION

.75 LOCATION `{in, based}` ORGANIZATION

先找出Name Entities，明確指出是哪一類Entity，再去分析他們之間的句子所使用的字
這樣就可以當作訓練資料，來預測其他Entity之間的關係

Distant Supervision

Snow, Jurafsky, Ng. 2005. Learning syntactic patterns for automatic hypernym discovery. NIPS 17
Fei Wu and Daniel S. Weld. 2007. Autonomously Semantifying Wikipeida. CIKM 2007
Mintz, Bills, Snow, Jurafsky. 2009. Distant supervision for relation extraction without labeled data. ACL09

- Combine bootstrapping with supervised learning
  - Instead of 5 seeds,
    - Use a large database to get huge # of seed examples
  - Create lots of features from all these examples
  - Combine in a supervised classifier

Bootstrapping + 大量的資料(supervised learning)

Dan Jurafsky

以大量資料為基礎去做bootstrapping，但並不會去遞迴擴充他的pattern

Distant supervision paradigm

- Like supervised classification:
  - Uses a classifier with lots of features
  - Supervised by detailed hand-created knowledge
  - Doesn't require iteratively expanding patterns
- Like unsupervised classification:
  - Uses very large amounts of unlabeled data
  - Not sensitive to genre issues in training corpus

Dan Jurafsky

原始資料包含relation以及該relation的seed tuples

Distantly supervised learning of relation extraction patterns

1. For each relation

   Born-In

2. For each tuple in big database

   <Edwin Hubble, Marshfield>
   <Albert Einstein, Ulm>

   把seed tuples丟回原本的big database找出句子

3. Find sentences in large corpus with both entities

   Hubble was born in Marshfield
   Einstein, born (1879), Ulm
   Hubble's birthplace in Marshfield

   分析句子的pattern，找出features

4. Extract frequent features (parse, words, etc)

   PER was born in LOC
   PER, born (XXXX), LOC
   PER's birthplace in LOC

   訓練supervised classifier

5. Train supervised classifier using thousands of patterns

$$P(\text{born-in} \mid f_1, f_2, f_3, \ldots, f_{70000})$$

不需要用patterns再去遞迴找其他資料，因為原始資料就已經夠多了

如果手邊沒有任何資料，那就先從網路上隨便抓
把抓下來的資料拿去訓練trustworthy tuple classifier（三人成虎）
–>如果有很多網站都說這個Name Entities pair relation是正確的，那他就是對的
以後如果找到Name Entities，想判斷他們究竟是什麼relation
就把Name Entities和relations丟進trustworthy tuple classifier去判斷並排序

Dan Jurafsky

Unsupervised relation extraction

Text runner Algorithm
(Banker Algorithm)

M. Banko, M. Cararella, S. Soderland, M. Broadhead, and O. Etzioni. 2007. Open information extraction from the web. IJCAI

- Open Information Extraction:
  - extract relations from the web with no training data, no list of relations

1. Use parsed data to train a "trustworthy tuple" classifier
2. Single-pass extract all relations between NPs, keep if trustworthy
3. Assessor ranks relations based on text redundancy

(FCI, specializes in, software development)

(Tesla, invented, coil transformer)

Dan Jurafsky

Evaluation of Semi-supervised and Unsupervised Relation Extraction

- Since it extracts totally new relations from the web
  - There is no gold set of correct instances of relations! 因為不知道哪些relation是正確的
    - Can't compute precision (don't know which ones are correct)
    - Can't compute recall (don't know which ones were missed)
- Instead, we can approximate precision (only)
  - Draw a random sample of relations from output, check precision manually

$$\hat{P} = \frac{\text{\# of correctly extracted relations in the sample}}{\text{Total \# of extracted relations in the sample}}$$

等於是要人工計算precision

- Can also compute precision at different levels of recall.
  - Precision for top 1000 new relations, top 10,000 new relations, top 100,000
  - In each case taking a random sample of that set
- But no way to evaluate recall

我們不可能對網路上所有的relations做precision, recall計算
但可以抽出一些樣本來算precision

Relation Extraction

Semi-supervised and unsupervised relation extraction