COMP4650 / COMP6490
Document Analysis 2018

Information Extraction

Gabriela Ferraro

DATA 61 CSIRO

Australian National University

Overview of IE lectures

- Introduction to Information Extraction (IE)
  - Overview
  - Relation Extraction
  - Named Entity Recognition
- Sequence labeling methods 1 and 2
- Automatic Summarization

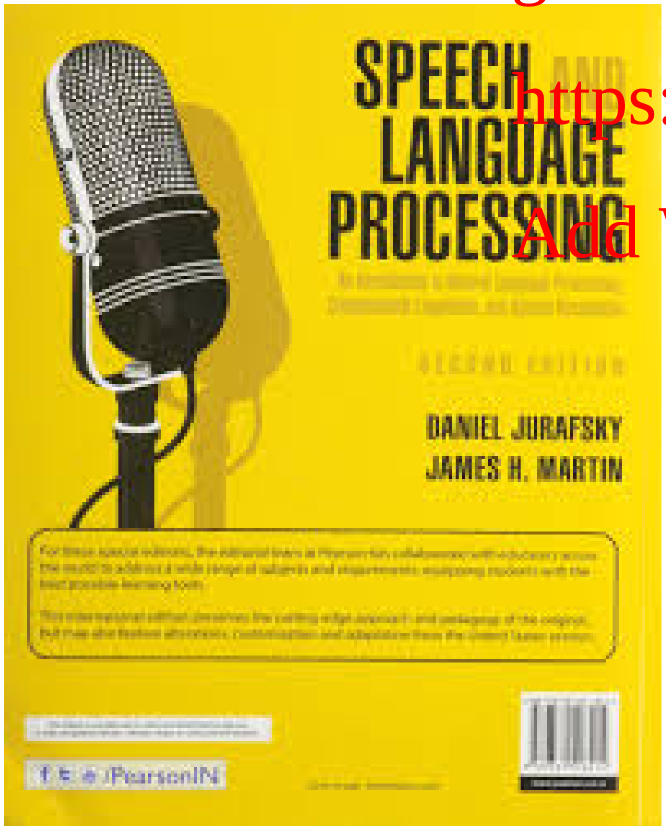
* Acknowledgement: Some of the content originates from the Stanford NLP course at Coursera.org

Books

Speech and Language Processing

Jurafsky and Martin

2014. Pearson.



Natural Language Processing

Jacob Eisenstein

2018. MIT pres.

https://github.com/jacobeisenstein/gt-nlp-class

Introduction to IE

What is IE?

Automatically extract structured information from unstructured and/or semi-structured data.

Who did what to whom when?

Main goals:

- Helps natural language understanding
- Organize information for humans
- Organize information in a formal and precise form that allows further analysis and/or inferences made by computer algorithms

IE Applications

Scan documents and populate:

- Templates
- Ontologies
- Data Bases
- Knowledge Bases

Text understanding (e.g.: named entity recognition, relation extraction)

Automatic summarization

Question answering

...

etc.

IE Template based example

2008 January 17
British Airways Flight 38, a **Boeing 777- 200ER**, lands short of the runway at **London Heathrow Airport** in the United Kingdom. **Nine** of the 152 people on board are treated for minor injuries, but there are **no fatalities**; this is the first loss of a Boeing 777.

Template

Type of Information to be extracted

SLOT	VALUE
DATE	17/01/2008
FLIGHT#	38
AIRCRAFT	Boeing 777-200ER
COMPANY	British Airways
ORIGIN	
DESTINATION	London Heathrow, UK
VICTIMS	0
INJURED	9

Extracted information

Extract information about aircraft accidents from news

Templates types

Slots in a template are usually filled by a substring of a document

- Some slots may have a **fixed set of fillers**


```
Job type: nurse | doctor | physic
```


- Some slots may allow **multiple fillers**

```
Programming language: Java, C++, Python, etc.
```

IE applications

- **Relation Extraction**

  Paris **is the capital** of France.

  France's **capital** is Paris.

  Paris <is-a-capital-of> France

- **Name Entity Recognition and classification**

  Paris is the capital of France.

  <LocationEntity> <...> <LocationEntity>

- **Combined**

  <LocationEntity> <is-a-capital-of> <LocationEntity>

IE methods

Hand written patters

Supervised machine learning

Semi-supervised and unsupervised learning

Relation Extraction

How relations are express in natural language?

- Relations are instantiated by predicates
- Predicates have arguments
- Verbs are the most productive predicate form


```
predicate(arg1, args2, ..., argn)
```

*Mery **likes** cake.*

```
likes(Mery, cake)
```

*Mery **rent** a boat for 2 weeks for 300 dolars.*

```
rent(Mery, boat, 2 weeks, 300 dolars)
```

Why Relation Extraction?

Create new structured knowledge, e.g., facts

- Augment current knowledge bases

  Adding words to WordNet thesaurus, facts to FreeBase or DBPedia

- Support question answering

  `Which actor starred in the film BATMAN 3?`

  acted-in(?x, BATMAN)

  is-a(?y, actor)

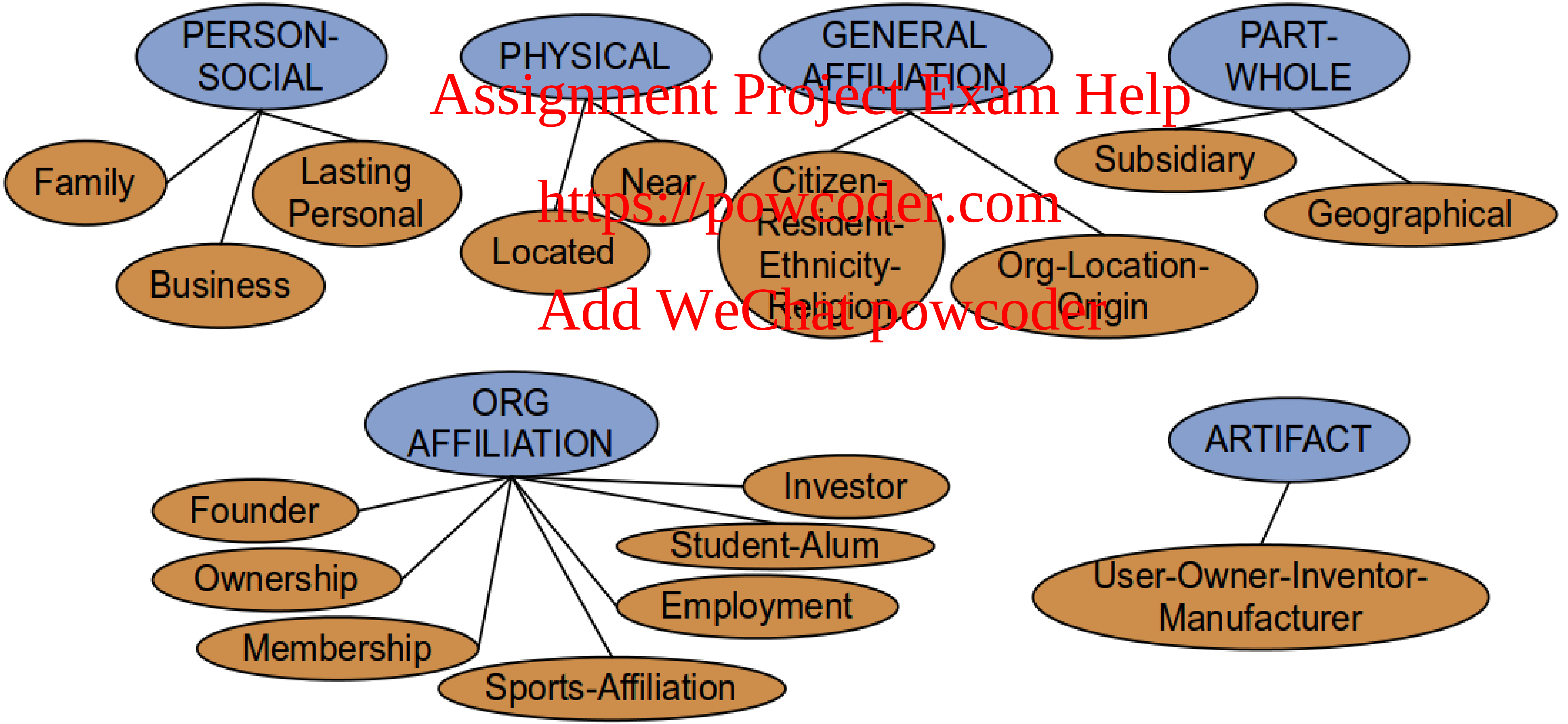
But which relations should we extract? And how?

Which relations to extract?

- A pre-defined set of relations
- All relations (e.g., all verbs and their arguments)
- Ontological relations

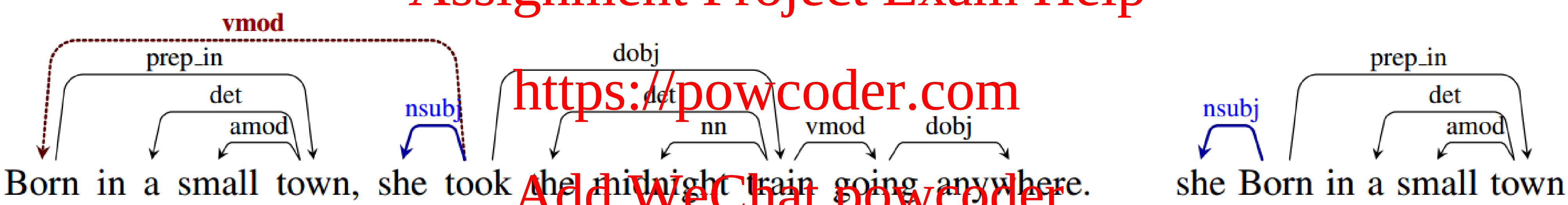
Example of a pre-defined set of relations

17 relations from SemE 2008 "Relation Extraction Task



Example of extracting ALL relations

Use sytactic dependency trees to extract predicates and their arguments



(input) ↓ **(extracted clause)** ↓

she took the midnight train going anywhere
Born in a small town, she took the midnight train
Born in a town, she took the midnight train
she took the midnight train
she took midnight train
. . .

↓

(she; took; midnight train)

she Born in small town
she Born in a town
she Born in town

↓

(she; born in; small town)
(she; born in; town)

Ontological relations

Examples from the WordNet Thesaurus http://wordnetweb.princeton.edu/perl/webwn

Hypernym (is-a): subsumption between classes

- Giraffe **IS--A** ruminant **IS--A** ungulate **IS--A** mammal **IS--A** vertebrate **IS--A** animal…

Hyponym relation or Instance-of: relation between individual and class

- Dog → Terrier → Bull-terrier → Staffordshire bull-terrier...

- San Francisco **instance-of** city

Synonym relation

- **Car** Sense 1 => auto, automobile, motorcar, machine
- **Man** Sense 1 => adult men
- **Man** Sense 2 => homo, human being, human

Relation extraction projects

Resource Description Framework (RDF) triples

```
Golden Gate Park location San Francisco
```

Dbpedia: +1 billion RDF triples http://dbpedia.org/


```
dbpedia:Golden_Gate_Park dbpedia--owl:location
dbpedia:San_Francisco
```

Freebase relations: well-known people, places, and things
https://www.freebase.com/

```
Total RDF riples: 2.1M
```

How to build relation extractors

Hand written patters

Supervised machine learning

Semi-supervised and unsupervised learning

Hand written rules: 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...

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In Proceedings of the 14th conference on Computational linguistics - Volume 2 (COLING '92), Vol. 2. Association for Computational Linguistics, Stroudsburg, PA, USA, 539-545.

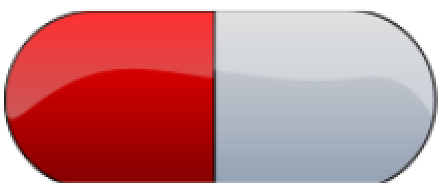
Extracting Richer Relations Using Rules

Intuition: relations often holds between specific entities

- **located-in**(ORGANIZATION, LOCATION)
- **founded**(PERSON, ORGANIZATION)
- **cures**(DRUG, DISEASE)

Start with Named Entity tags to help relation extraction

Which relations hold between 2 entities?



Drug

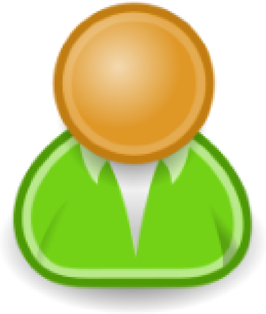
Cure?

Prevent?

Cause?

Disease

Which relations hold between 2 entities?



PERSON

Founder?

Investor?

Member?

Employee?

President?

ORGANIZATION

Summary: Hand written patterns for Relation Extraction

- **Plus**
  - Human patterns tend to be high-precision
  - Can be tailored to specific domains
- **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 would like better accuracy

Supervised machine learning for Relation Extraction

Training

- Choose the set of relations you want to extract
- Find and label data* = training set creation
- Extract relevant features from the training set
- Train a classifier on the training set

Testing

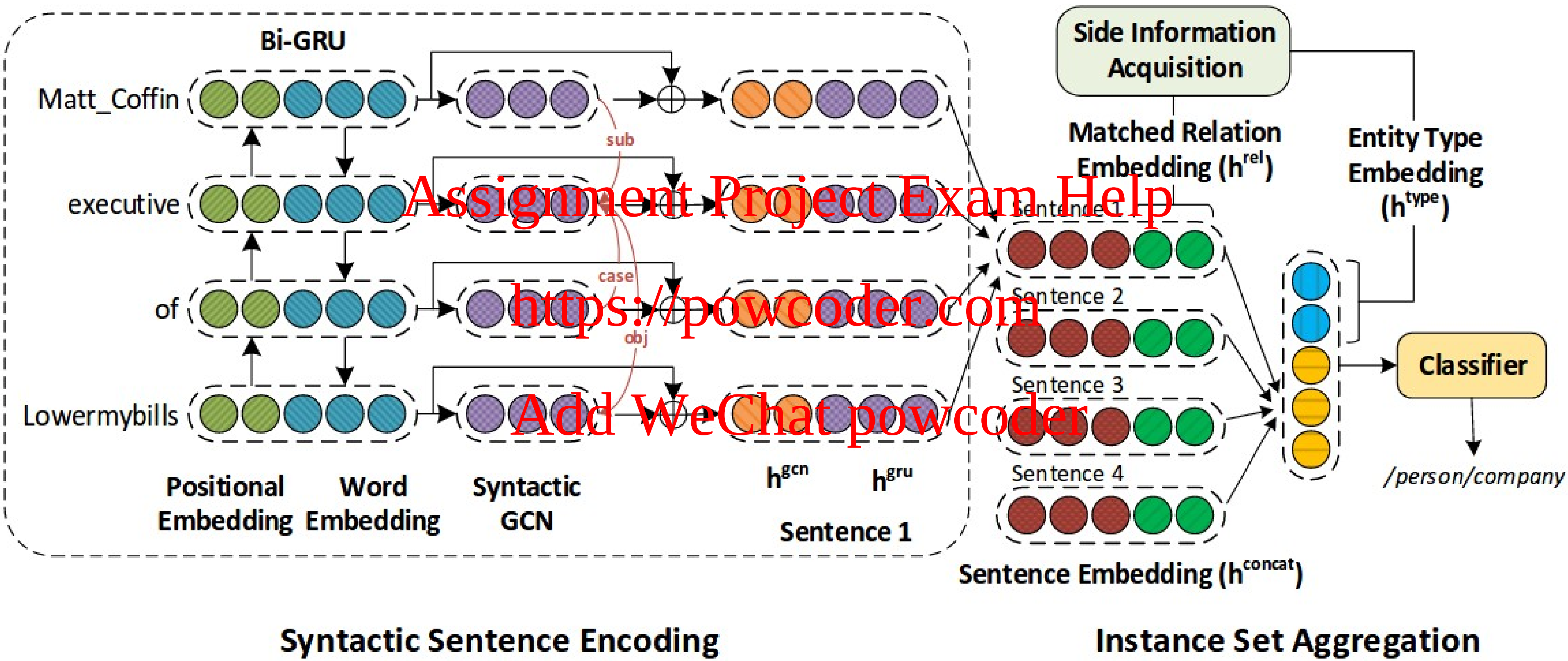
- Tuned the classifier parameters on the dev. set
- Test the classifier on the test set

* Available RE datasets: SemEval7; BioNLP, etc.

Supervised relation extraction between entities

- Find all pairs of named entities (`person, location, organization`)
  - Decide if 2 entities are related
  - If yes, classify the relation into relation types (`is-a, instance-of, born-in`, `etc`.)

- You can use any classifier you like
  - MaxEnt, Naive Bayes, CRF, SVM, CNN, etc.



RESIDE: Improving Distantly-Supervised Neural Relation Extraction using Side Information

(Vashishth et al., 2018)

Summary: Supervised machine learning for Relation Extraction

- **Plus**
  - Can get high accuracy with enough hand-labeled training data, if test data is similar enough to training data

- **Minus**
  - Labeling a large training set is expensive
  - Supervised models are brittle, don't generalize well to different genres

Semi supervised Relation Extraction

No training set? Maybe you have:

- A few **seed** tuples
- A few high-precision patterns

Can you use those **seeds** to do something useful?

- Bootstrapping: use the seeds to directly learn to populate a relation

Relation Bootstrapping (Hearst, 1992)

- Gather a set of ***seed pairs*** that have relation ***R***
- Iterate:

  - Find sentences with these pairs

  - Look at the context between or around the pair and generalize the context to create patterns
  - Use the patterns for *grep* for more pairs

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

Unsupervised relation extraction

- Extract relations from with no training data, thus no pre-defined list of relations

- Single-past: extract all relations between NPs

- Assessor ranks relations based on text redundancy

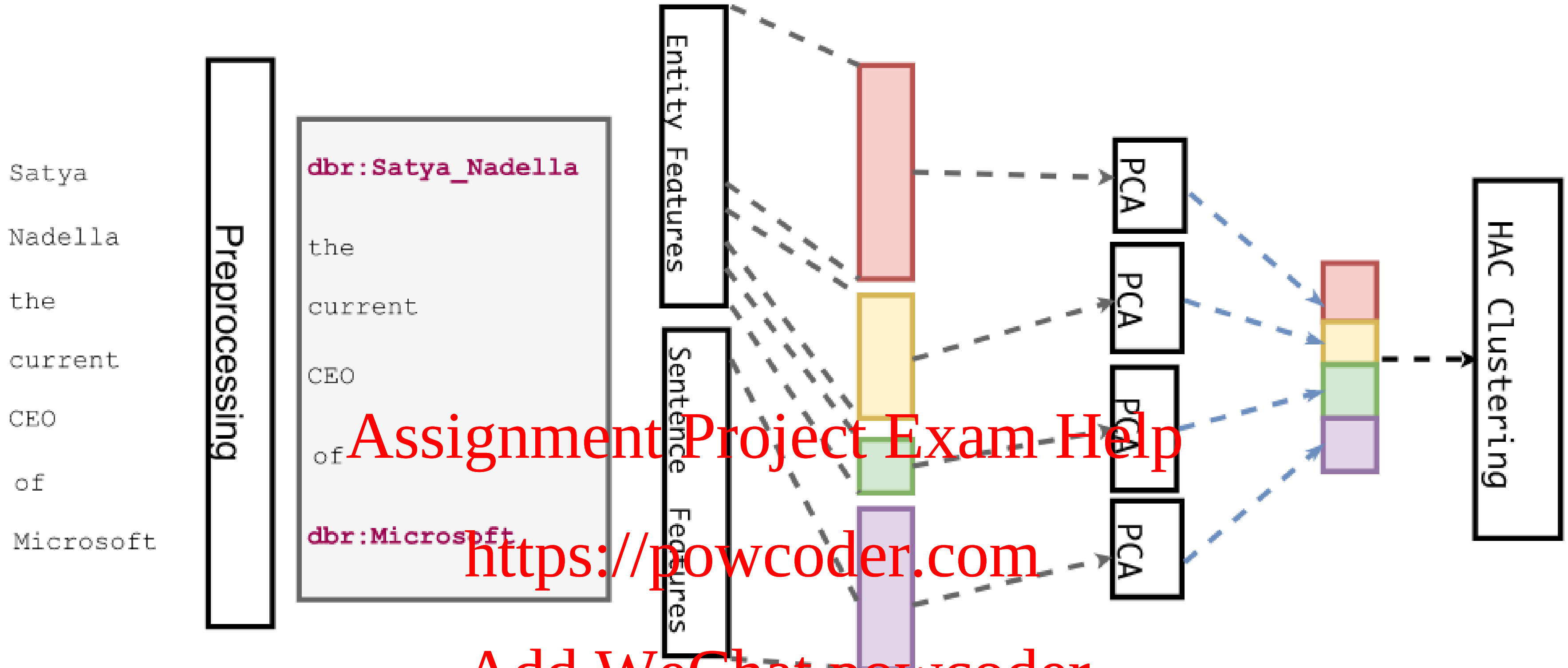


Fig. 2: System overview

Preprocessing For each sentence in the dataset, we extract named entities using DBpedia Spotlight and consider all sentences containing at least two entities. For this set of sentences, the Stanford CoreNLP dependency parser is utilized to extract the lexicalized dependency path between each pair of named entities.

Elsahar et al., 2018

Evaluation of unsupervised relation extraction

- Since it extracts totally new relations...

  there is no gold set of correct relations

  - cannot compute precision (don't know which ones are correct)
  - cannot compute recall (don't know which ones were missed)

- Instead, we can approximate **precision**

  draw a random sample of relation from output, check precision manually

Name Entity Recognition

Name Entity Recognition

Named Entity Recognition (NER)
Find and classify names in texts,
e.g.: person, location, organization, number, currency, etc.

Designated **S/2004 N 1**, this is the **14th** known moon to circle the giant planet.

It also appears to be the smallest moon in the **Neptunian** system, measuring just **20 km** (**12 miles**) across, completing one revolution around **Neptune** every **23 hours**.

US astronomer **Mark Showalter** spotted the tiny dot while studying segments of rings around Neptune.

proper name
quantity
location
person
Time
Other

NER Applications

- Machine Translation
- Question Answering
- Automatic Summarization
- Relation Extraction

NER as learning

- **Training**
  - Collect a set of representative training documents
  - Label each token for its entity class or other
  - Design feature extractors appropriate to the text and classes
  - Train a sequence classifier to predict the labels from the data
- **Testing**
  - Receive a set of testing documents
  - Run sequence model inference to label each token
  - Appropriately output the recognized entities

NER Task: the training data

US	**LOC**
astronomer	**O**
Mark	**PER**
Showalter	**PER**
spotted	**O**
that	**O**
:	**:**

Standard evaluation is per Entity not per token

Precision, recall and F-measure

NER Task: example features

Numbers

- twoDigitNum (90) = Two-digit year
- fourDigitNum (1990) = Four-digit year
- containsDigitAndAlpha (A8956-67) = Product code
- containsDigitAndDash (09-96) = Date
- containsDigitAndSlash (11/9/89) =Date
- containsDigitAndComma (23,000.00) = Monetary amount
- containsDigitAndPeriod (1.00) = Monetary amount, percentage

NER Task: example features

- Person
  - capPeriod (M.) = Person name initial
  - initCap (Sally) = Capitalized word
  - lowerCase (can) = Uncapitalized word

- Organization
  - allCaps (IBM) = Organization

* Gazzeters (list with persons, organizations, abbreviations, etc.)

NER challenges

Ambiguity problems:

- **Paris** (city vs. person)
- **May** (person vs. month)
- **2013** (date vs. quantity)
- **Ferrari** (person vs. organization)

Multi-language NER:

- Language independent features (position, suffix, prefix, digits, POS-tags)
- Lack of capitalization (Chinese, Indian lang., etc.)
- Too much capitalization (German)
- Free word order languages (Hungarian, Russian, etc.)
- Languages with rich morphology (Czech, Spanish, etc.)

Evaluation in IE

How much relevant information has been extracted

Precision = # of correct answers given by the system /

total # of possible correct answers in the text

How much of the extracted information is correct

Recall = # of correct answers given by the system /

# of answers given by the system

How good is the system in ignoring spurious information

Fall out = # of incorrect answers given by the system /

# of spurious facts in the text

Combination of Precision and Recall

F-Measure = 2 * (Precision * Recall / Precision + Recall)

IE take away

IE deals with processing human language texts by means of natural language processing techniques

- **Rule based methods**
  - Use lexical patterns, e.g.: `X was born in Y`;
  - Use syntactic patters, e.g.: `Subject, Verb, Object`
- **Supervised methods**
  - Sequential labeling algorithms as HMM, MME, CRF
  - Required training data
- **Semi-supervised and unsupervised methods**
  - Semi: required seed examples, e.g. lexical patterns
  - Unsupervised: require unlabeled data
  - Evaluation is not straightforward

Conclusion

- In the future, IE from cross-website pages will become more important as we move towards the Semantic Web

- IE new challenges are: domain independent solutions, data integration and multilingualism
  - Lots need to be done!

Resources/Tools

KnowItAll

https://github.com/knowitall

Stanford Named Entity Recognizer (Lafferty, McCallum, and Pereira, 2001)

http://nlp.stanford.edu/software/CRF-NER.shtml

OpenIE

https://nlp.stanford.edu/software/openie.html