Text Mining CBS 2019

Lecture 6: Relation extraction

Piek Vossen

VU UNIVERSITY AMSTERDAM

Literature

- NLTK book, chapter 7, section 6: https://www.nltk.org/book/ch07.html
- Background reading:
  - T. Mitchell et al. (2015) Never-Ending Learning, Association for the Advancement of Artificial Intelligence (www.aaai.org)
  - Kozareva, Z., Riloff, E., & Hovy, E. H. (2008). Semantic Class Learning from the Web with Hyponym Pattern Linkage Graphs

Overview of relation extraction

- Relation extraction
  - between instances (property-value extraction or knowledge base completion):
    - Microsoft ***hasCEO*** Bill_Gates, Bill_Gates ***isA*** CEO
    - Definition of the task
    - Different Approaches
    - Performance
  - between concepts (ontology learning)
    - CEO ***subclassOf*** Manager, Manager ***subclassOf*** person
    - Company ***make*** Product, Pizzeria ***subclassOf*** Company, Pizza ***subclassOf*** Product
    - Different approaches

Definition of the task of property-value extraction

- A relation is mostly a triple
    - **SUBJECT - PROPERTY/RELATION - VALUE/OBJECT**
    - RDF, but also columns in relational database
    - Barack_Obama **hasAge** 57;
    - Barack_Obama **hasPosition** president

Definition of the task of property-value extraction

- 3 subtasks
  - identifying the arguments
  - interpreting the arguments
  - detecting the property or relation
- **Barack Hussein Obama II is** an American **attorney** and **politician** who **served as** the 44th **president** of the United States from 2009 to 2017. **Obama** was **born** in **Honolulu**, Hawaii. After **graduating** from **Columbia University** in **1983**, he **worked as a community organizer** in Chicago. https://en.wikipedia.org/wiki/Barack_Obama
- **Former President Barack Obama turned 57** on Saturday — and for the first time, his birthday is being celebrated as a commemorative holiday in his former home of Illinois. http://time.com/5358013/barack-obama-birthday-57/

Property-value extraction builds on top of NERC and NED pipelines

- Knowing the entities you can almost guess the relation
  - Trump - 71, Barack - 57, Merkel - 63, Clinton (which one?)- 70
  - Trump - US, Merkel - Germany, Gates - Microsoft, Messi - Barcelona
- But not always:
  - Al **Gore**, Frank **Gore** (American Football), Lesley **Gore** (singer, songwriter, actress, activist, 68, died in 2015), **Gore** Vidal (writer, 80)
  - Trump - Emma Brockes ***OR*** Gore Vidal - 71
  - Clinton - Clinton

Ambiguity, variation and vagueness

- **Variation**: ***was born on***, ***birthdate is on*** or ***saw the light of day for the first time on***.
- **Ambiguity**: "*works at*" can imply employeeOf or CEO; ***Ford*** can be a car, a company or a president; ***Jaguar*** can be an animal or a car; who is ***John Smith***?
- **Vague** relation expressions: ***Alfred Hitchcock's 'The Birds' was very popular***.
  - **Knowing** that Alfred Hitchcock is an <u>author</u> **helps** concluding that *The Birds* is likely a <u>book</u> but if he is a <u>director</u> it is perhaps a <u>film</u>
- Relations can also span **several sentences** or contain **indirect references**:
  - In November 1963 ***Capitol Records*** finally signed a contract with ***the Beatles*** and ***announced*** plans to ***release*** the Beatles' single 'I Want To Hold Your Hand' in December 1963 as well as ***their second album*** With The Beatles in January.

second album of Capital Records or the Beatles?

Different task definitions

- Sentence-level extraction (text annotation) versus instance-level extraction (triples)
- Closed property-value extraction:
  - Slot filling or template-based: no semantic constraints on the type of object/value
  - Schema-based relation extraction: subjects, objects and properties/relations are pre-defined in an ontology
- Open information extraction
- Knowledge base population/completion

Property-value extraction in text

- Text annotation: text segment in which a specific relation is being expressed, e.g. "[John Smith <PER>] was [born <place-of-birth>] in [Chicago <LOC>]".

- Slot filling: given a relation schema, extract the slots and relation from text

  - Message Understanding Conferences (MUC): https://cs.nyu.edu/cs/faculty/grishman/muc6.html

  - Knowledge Base Completion (TAC-KBC): https://tac.nist.gov

Closed Relation Extraction & schema definitions

- predefined list of relations, e.g. <found-of>
- number of arguments (usually binary (RDF triples), but also: *buyer-buys-goods-from-seller*)
- type of slot fillers: PERSON, <founder-of>, ORGANISATION
- Rang of fillers:
  - only one answer: <birth-place>, <birth-date>
  - two answers: <child-of>
  - open list answers: <married-to>, <founder-of>

Schemas

- YAGO:
  - Knowledge Base derived from Wikipedia, WordNet and GeoNames: 10 million entities (like persons, organizations, cities, etc.) and contains more than 120 million facts about these entities.
  - https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/
- http://schema.org: vocabularies to capture entities, relationships between entities and actions: http://schema.org/Movie
  - <div **itemscope itemtype** ="**http://schema.org/Movie**">
  - <h1 **itemprop**="**name**">Avatar</h1>
  - Science fiction
  - </div>

Advantages of using a schema

- **Disjunct properties**: ***Ford*** cannot be a person, company and a car at the same time while only one has a birth-date
- **Property dependencies** & **correlations**: Filip Ilievski's profiler learns from DBPedia,
  - An American politician, member of the Senate. Probably a middle-aged white man with a degree. Where did he get his degree, and what is his religion?
- **Constraints** on fillers to semantics types: certain relations are limited to politicians, football players, movie directors, etc.
- **Generalise** data from instances, e.g. if some movies have directors than also other movies are likely to have directors.
- Semantic web and Linked Open Data:
  - **Class hierarchies** (ontologies): all companies, countries, politicians, directors, writers
  - **Unique Resource Identifiers** (URIs): different identities for ***Ford***

Methods

- Rules and patterns, Supervised, Unsupervised (clustering)
- Bootstrap: start with seed relations to learn patterns and obtain new seeds, etc…
- Distant supervision: get seed data from a knowledge base
- Closed Relation Extraction: there is a schema or a fixed set of known relations
- Open Relation Extraction: there is no schema or predefined list of relations

Using Regular Expressions

- Which PERSON holds what POSITION in what ORGANISATION
  - [PER], [POSITION] of [ORG]
    [ORG] (named, appointed, …) [PER] Prep [POSITION]
  - Nokia has appointed Rajeev Suri as President
- Where is an ORGANISATION located
  - [ORG] headquarters in [LOC]
  - NATO headquarters in Brussels
  - [ORG] [LOC] (division, branch, headquarters…)
  - KFOR Kosovo headquarters

Using Regular Expressions

FASTUS

- Appelt, D.E., Hobbs, J.E., Bear, J., Israel, D. & Tyson, M. (1993), FASTUS: A finite-state processor for information extraction from real world text. IJCAI, pp.1172-1178.
- Cascade of finite state machines (small programs with if-then statements that can be combined)
- Regular expressions over text sequences: words, part-of-speech, constituent heads, syntactic role, semantic types template filling:
  - MUC-4: 44 recall & 55 precision (1992)
  - MUC-6: 74 recall & 76 precision (3 years later)

Slot/template filling task in the Message Understanding Conference (MUC)

'The mayor's home was attacked by terrorists.'
'Terrorists attacked the mayor's home in Bogota.'
'The home of the mayor of Bogota suffered a grenade attack.'

Table 3.3 FASTUS extraction template for the terrorist domain

Field	**Filler**
MESSAGE ID	TST-MUC3-0002
DATE OF INCIDENT	04 FEB 90
TYPE OF INCIDENT	ATTACK
PERPETRATOR	TERRORISTS
PHYSICAL TARGET	HOME
HUMAN TARGET	MAYOR
INSTRUMENT	...
LOCATION OF INCIDENT	BOGOTA

grenades

Bootstrapping (DIPRE, KnowItAll but also NELL)

- Seed relations & web search (or equal size of text index)
- Double-propagation (similar as for sentiment-target extraction):
  - Using books you find patterns, using patterns you find books, using new books you find new patterns, etc….
  - <Subject, R, Object>; 5 examples, “Tolkien” <authorOf> “The lord of the rings”
  - Text search for occurrences of S and O: “Tolkien” + ”The lord of the rings”
  - Extract patterns from the results by replacing S & O by variables:
    - *The Lord of the Rings is a book by J.R.R. Tolkien*
    - Pattern: O is a book by S
  - Results for S, O are used to search the web for more phrases

Bootstrapping (DIPRE, KnowItAll but also NELL)

- Exploit **redundancy** on the web (same relation is found on multiple locations):
    - Pointwise mutual information score
        - N(Entity1 relation Entity2) / N(Entity1) * N(Entity2)
- **Specificity**: not too many hits for a pattern & more than the seed for a pattern

Never Ending Language Learning (NELL), Tom Mitchell, Carnegie Mellon University

- Semi-supervised learning method runs 24/7 to train hundreds of different extraction methods for a wide range of categories and relations.
- Accumulated a knowledge base of 3,109,311 asserted instances of 1,186 different categories and relations, since 2010
- Unfortunately offline now: http://rtw.ml.cmu.edu/rtw/

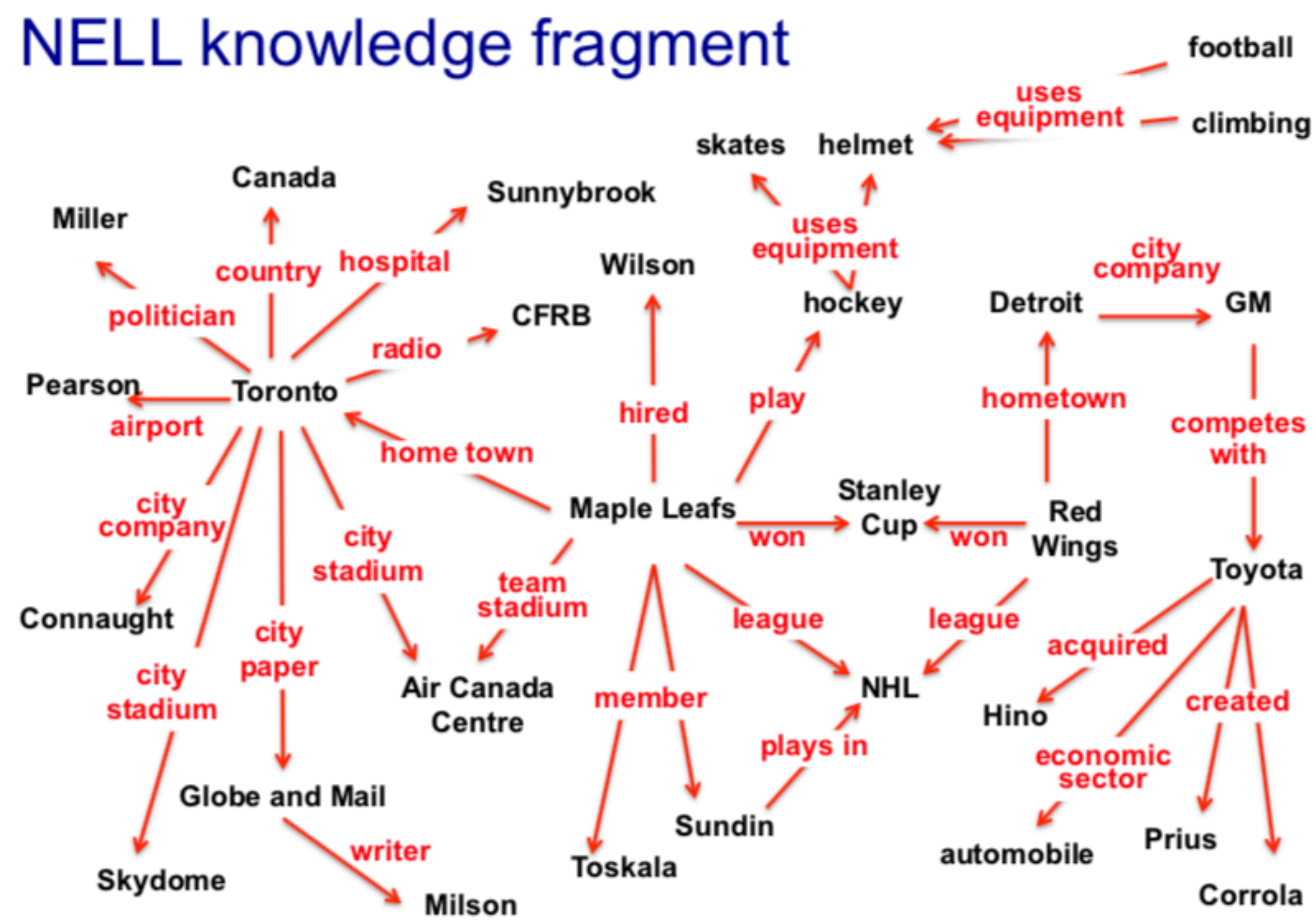


Figure 1: **Fragment of the 80 million beliefs NELL has read from the web.** Each edge represents a belief triple (e.g., play(MapleLeafs, hockey)), with an associated confidence and provenance not shown here. This figure contains only correct beliefs from NELL's KB – it has many incorrect beliefs as well since NELL is still learning.

Never Ending Language Learning (NELL), Tom Mitchell, Carnegie Mellon University

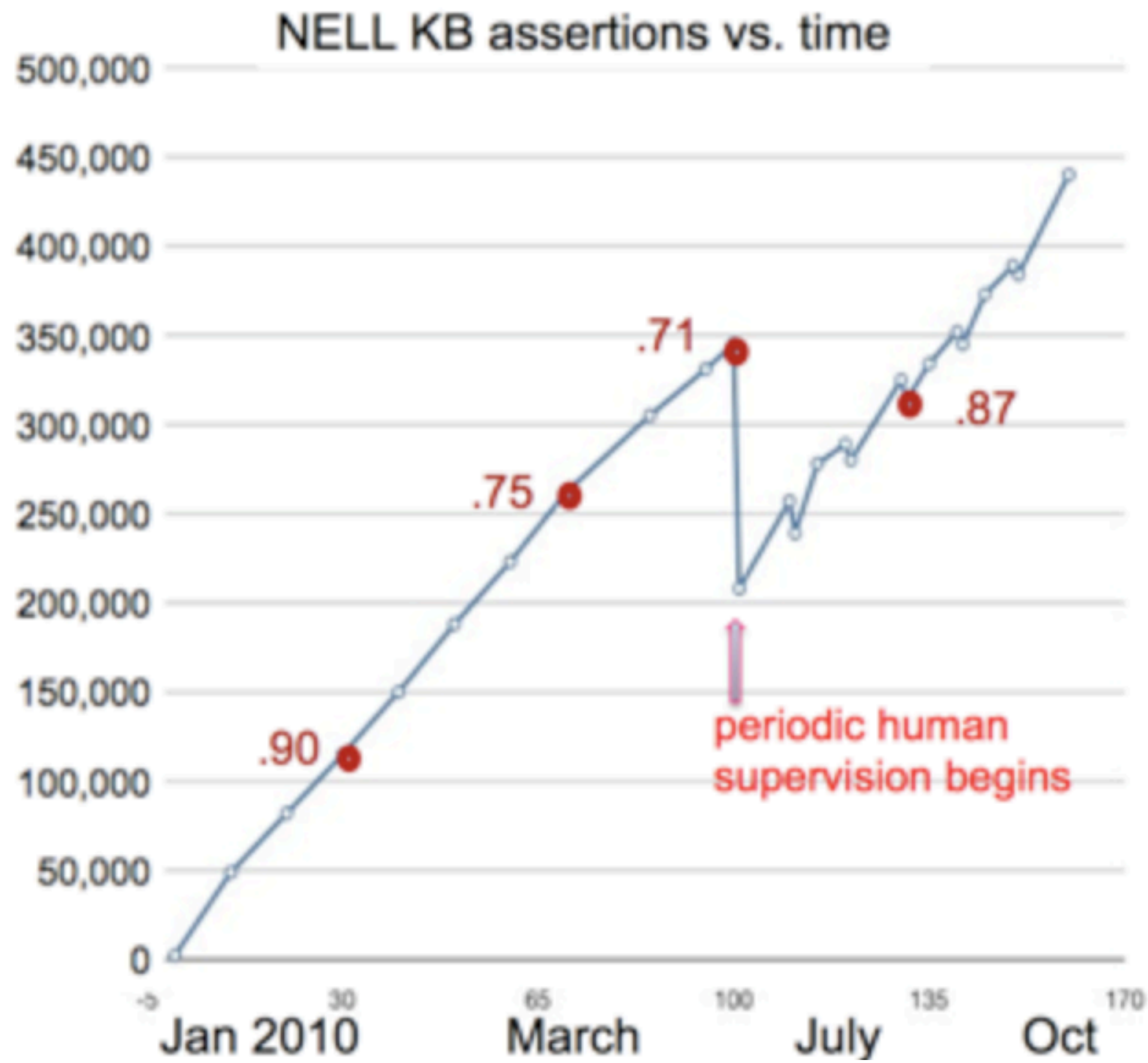
- Initial ontology with hundreds of categories (e.g., person, sportsTeam, fruit, emotion) and relations (e.g., playsOnTeam (athlete ,sportsTeam), playsInstrument (musician, instrument)) and 10 to 15 seed examples.
- Training of relation detectors on text and tables for the given relations using variety of features and exploiting redundancy on the web
- Index 500 million web pages and access to the remainder of the web through search engine APIs,

Never Ending Language Learning (NELL),
Tom Mitchell, Carnegie Mellon University

- Find pairs of noun phrases for input relations (e.g., the pair "Jason Giambi" and "Yankees" is an instance of the ***playsOnTeam*** relation) and applies high-precision detectors
- New instances are added to knowledge base & detection methods are retrained using the new examples.

Concept Drift

- Precision declines over time
- Drift of bootstrapping methods, e.g. “work” can mean many different things (function, have an effect, provoke) and noise will change the meaning of “work” based on the data.
- Human intervention (active learning): NELL every few weeks, 5 minutes fix blatant errors



Supervised relation extraction pipeline

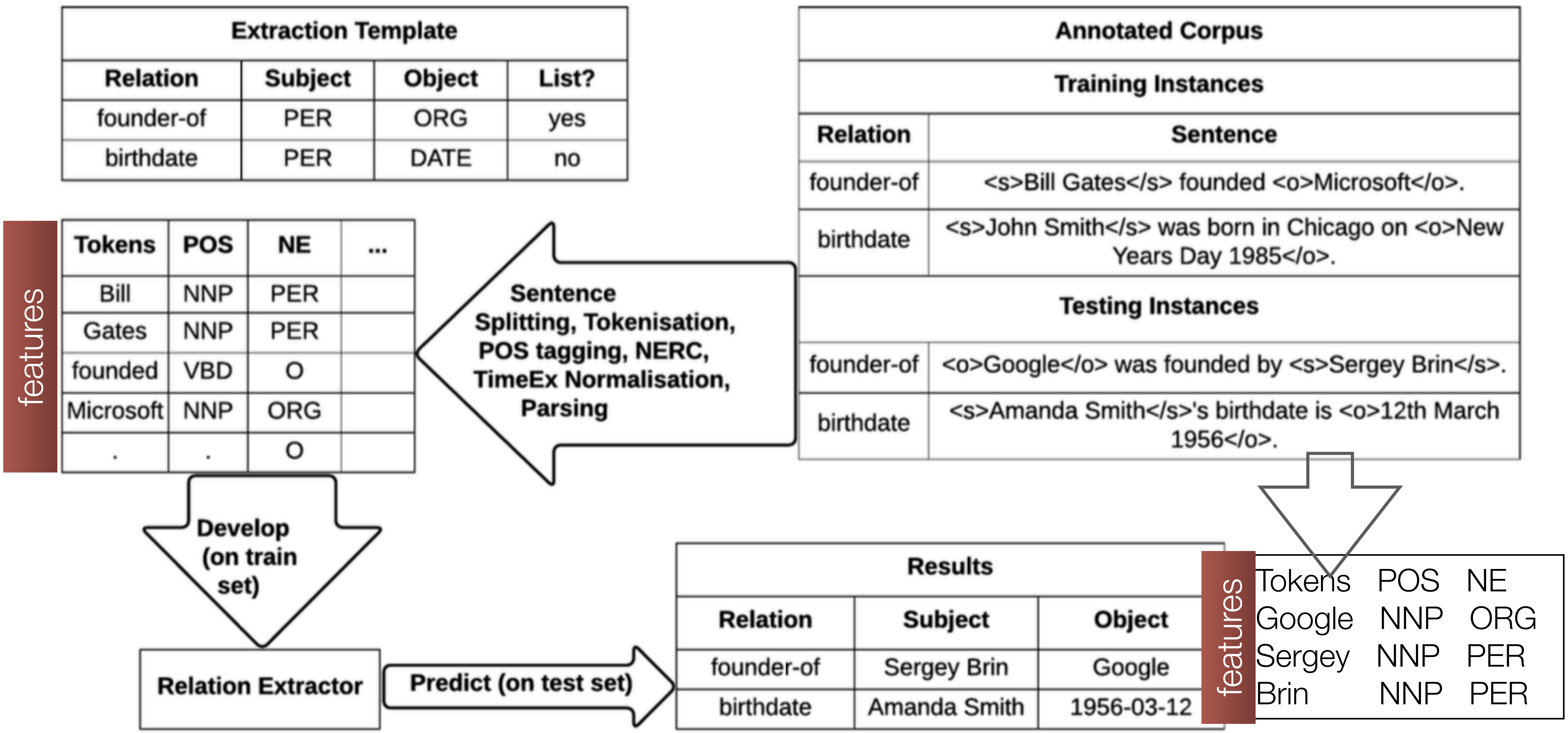


Figure 4.1: Typical Relation Extraction Pipeline

Feature sets need to match across training & test

Distant supervision

If two entities participate in a relation, any sentence that contains those two entities might express that relation.

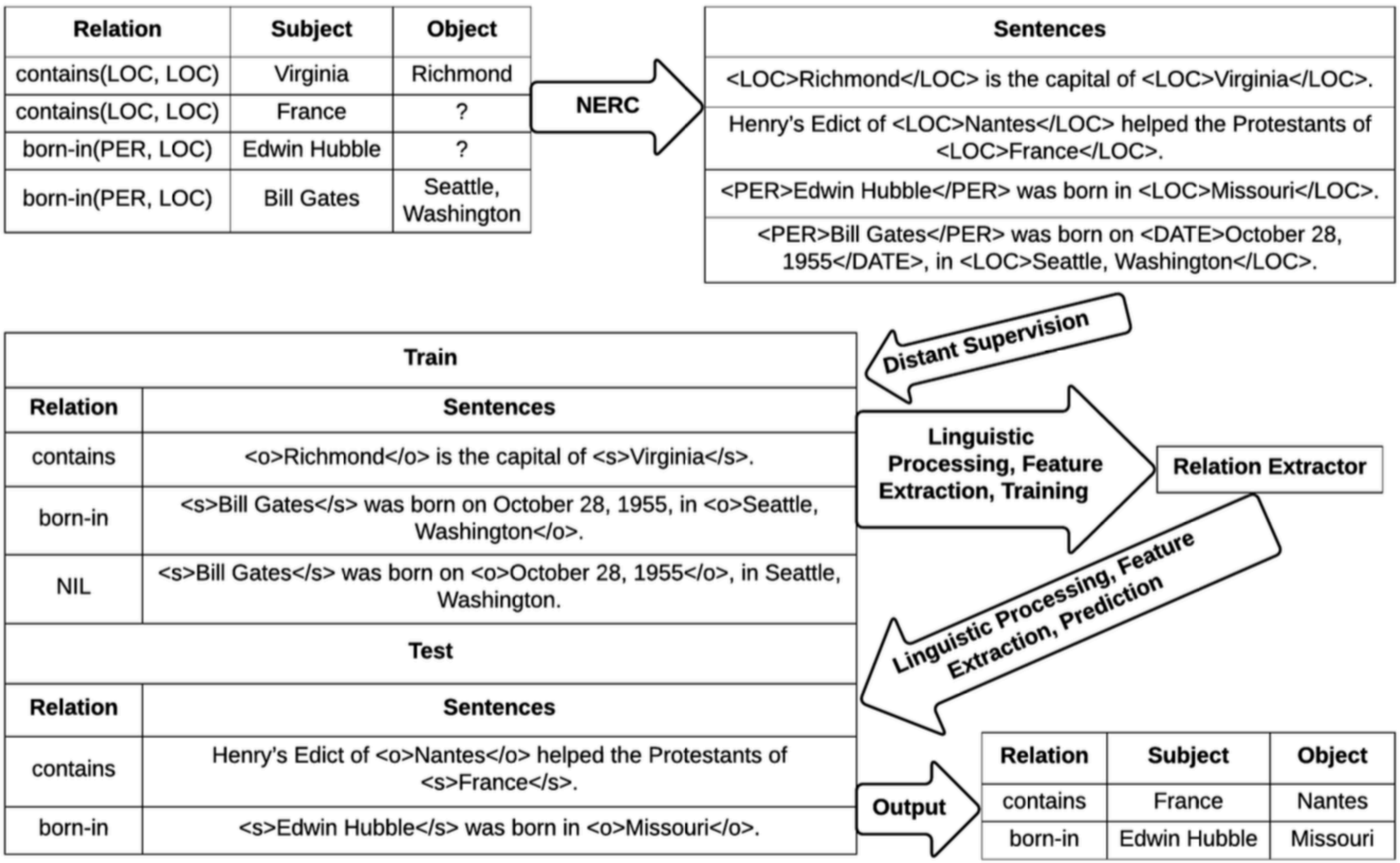


Figure 4.2: [81] Distant Supervision Method Overview

Training data for free

Building models (SVM, Max Entropy, Markov logic, CRF, NN)

- Feature sets:
  - N-grams of words/POS to left and right of entities;
  - Flag indicating which entity came first in sentence;
  - Sequence of POS tags and bag of words (BOW) between the two entities;
  - Syntactic dependency path between subject and object;
  - Lemmas and POS tags of words on the syntactic dependency path between the two entities;
- Relation embeddings and word embeddings in Neural networks —> capture latent features; less feature engineering needed
- Associate features with positive and negative pairs e.g. extracted from Knowledge Base
- Be aware of knowledge incompletion for negative pairs
  - Microsoft founded-by Bill_Gates may be missing in a Knowledge Base
  - ***absence of evidence is not evidence of absence***

Open Information Extraction

- Relations are not predefined but to be discovered from large collections of text: M1<ORG> <***relatedTo***> M2<PER>
- Double processing:
  - Process all data to annotate POS and NP Chunks
  - Supervised classifier trained on small subset with relevant/not-relevant or positive/negative cases
- Unsupervised clustering of sentences in which similar entities (embeddings) co-occur:
  - [Trump, Obama, Bush, Clinton] <***relatedTo***> [White House, government]
- Map detected clusters of relations to schema a posteriori —> <***worksAt***>
- Incoherent (contains X and omits Y) and uninformative relations (X has colour Y): syntactic constraints, sufficient variety of arguments, etc..

Overview of relation extraction methods

Method	Input	Output	Description	Advantages	Disadvantages
Boot-strapping	Unlabelled text, relation schema, rules and/or examples	Extraction rules, relations	Using a small set of extraction rules, extract examples, keep prominent ones, iteratively learn more extraction rule and examples	Easy to add new rules, can also be supplied by user	Often low recall and/or manual refinement needed for high precision
Rule-based	Unlabelled text, relation schema, rules and NE gazetteers	Relations	Using extraction rules and gazetteers, extract relations	Easy to add new rules, can also be supplied by user	Often low recall, much manual effort to develop
Supervised	Labelled text, relation schema	Relations	Using a schema and labelled training data, train model	Currently highest precision and recall for schema-specific relation extraction	Up-front effort of labelling data, risk of overfitting training set
Open IE	Unlabelled text	Groups of relations	Discover groups of relations from text using clustering, keep prominent ones	No knowledge about text needed	Difficult to make sense of groups and map to relation schemas
Distantly Supervised	Unlabelled text, relation schema, examples	Extraction model, relations	Using a schema and examples of relations, automatically annotate training data, train a model to extract more relations	Extracting relations with high recall and precision	Initial examples required
Universal Schema	Several partly populated knowledge bases	Unified knowledge	Take several KBs defined by different schemas, partly populated with relations, predict union of KBs	Integrate relations defined by different schemas after extraction	For small KBs it can be faster to do this manually

Table 4.2: Comparison of different minimally supervised relation extraction methods

Best performance

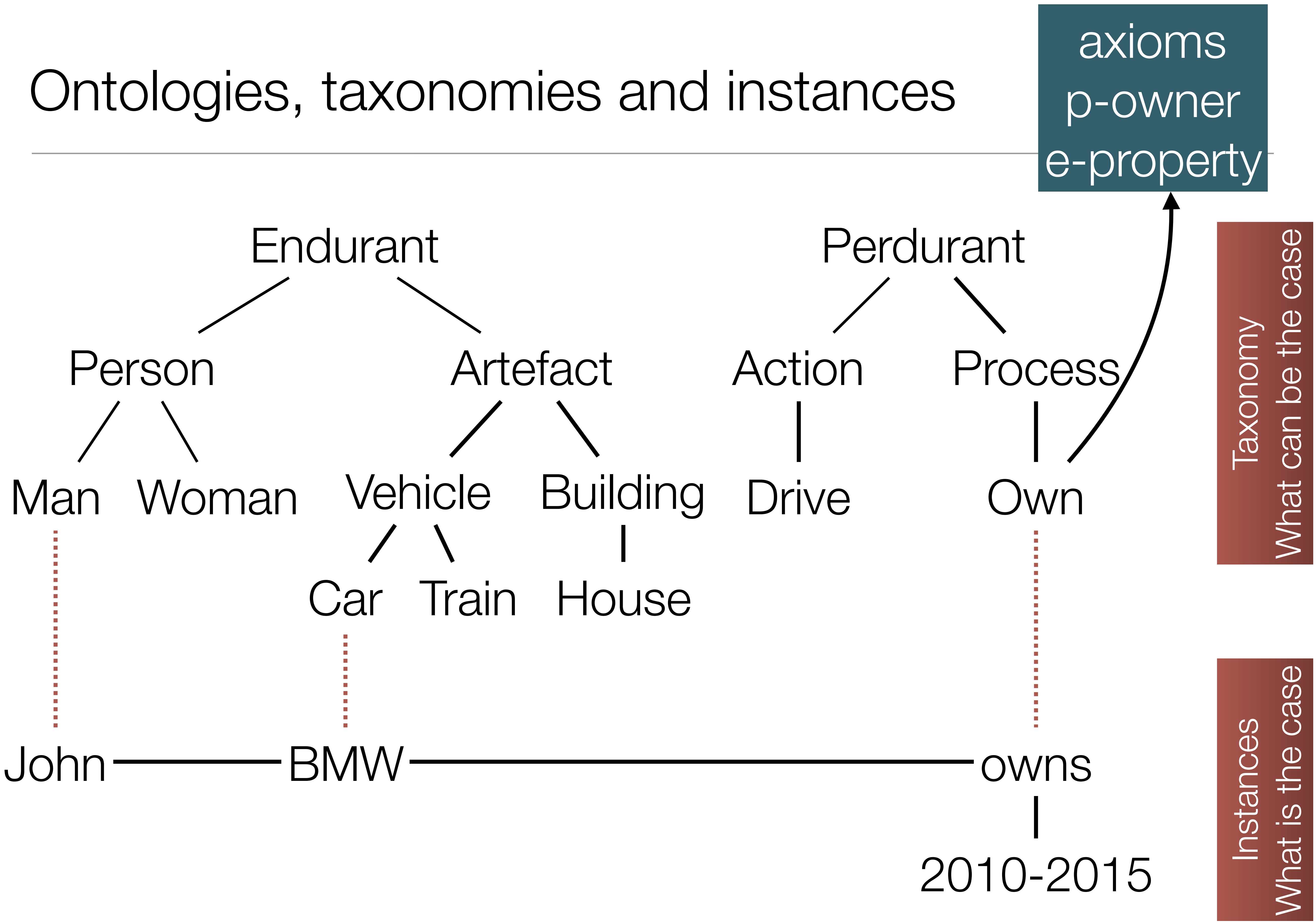
Trend

Performance per relation

- **Factors**:
  - Amount of training data
  - Quality of background data
  - Type of relation:
    - abstract - concrete
    - range of values
    - frequency of the relation
  - Popularity and redundancy
  - Range of possible relations (PER, ?R, PER)

Method	P	R	F1
employee of	32	46	38
top members	26	60	36
(org:)alt names	48	39	43
title	26	35	30
spouse	54	85	66
origin	43	70	53
cause of death	93	39	55
children	62	18	27
date of death	64	39	48
age	97	90	93

Ontologies, taxonomies and instances



Ontology Learning (OL) from text

- Differences with property-value extraction:
  - PV: non-taxonomic relations between instances (entities):
    - Trump <worksAt> White House
  - OL: taxonomic relations between concepts (nouns):
    - *apples* and *bananas* are *fruits; people workAt organisations*

Ontology Learning (OL) from text

- Why learn ontologies from text?
  - knowledge bases are incomplete, get out-of-date (products change, new products)
  - too expensive to build, revise and maintain manually
  - more empirical and captures cultural and temporal bound conceptualisations: ***pet***, ***toy***, ***weapon***, ***food***, ***drugs***
  - extremely valuable: reasoning, retrieval, generalising patterns and data from instances to types (fighting sparse data problem), as features in text representations for machine learning, embeddings in neural networks

Ontology learning cake

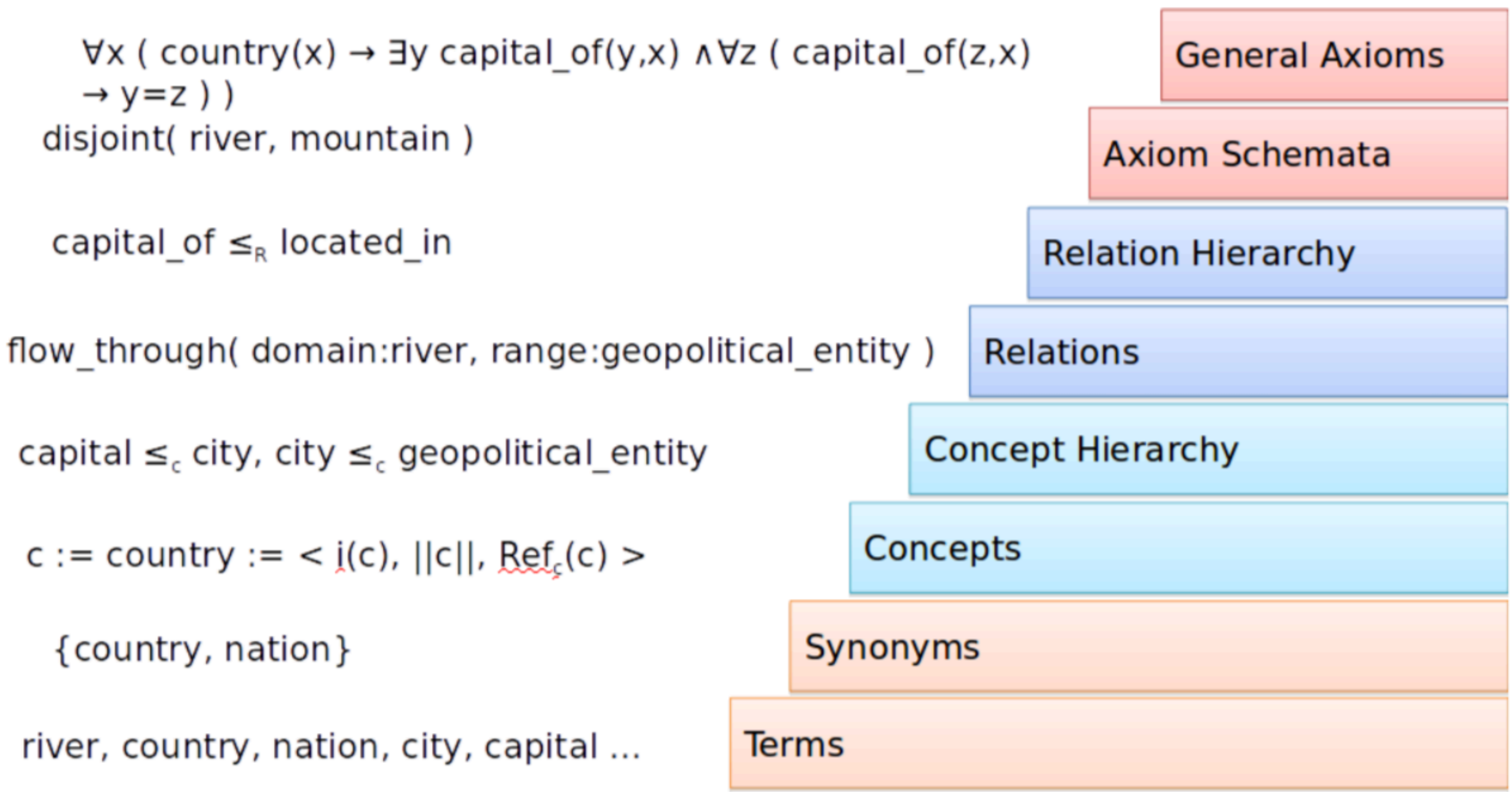


Figure 6.1: Ontology learning layer cake (reproduced from Cimiano, P: Ontology Learning and Population from Text: Algorithms, Evaluation and Application, Springer-Verlag, New York, 2006)

Methods

- What are the terms to learn?:

$$P(D_i \mid t) = \frac{count(t, D_i)}{count(t)},$$

  - Domain text **relevance**: words (single word terms) occurring significantly more frequent than expected (normalised term entropy)

  - **Co-occurrence** statistics: multiword terms "apple juice", "sailing boat", (e.g. point wise mutual information)

$$\operatorname{pmi}(x;y) \equiv \log \frac{p(x,y)}{p(x)p(y)}$$

- What is the relation between terms:

  - **Similarity** & **relatedness**: distributional models or word embeddings

  - **Hyponymy** & **meronymy**: Logical patterns for subtype or isa relations (manually or automatically learned)

What terms to learn?

Multiword terms

- Phrases
  - NN, AN, N of N, [A] N Preposition [A] N
  - *apple juice (NN), heavy metal (AN), frequently asked questions (BVN), toxic medication (AN), medication for toxication (NpN)*
- Co-occurrence statistics
  - Pointwise Mutual Information (pmi)
  - Association Ratio, Jaccard, Log likelihood

$$\mathrm{pmi}(x;y) \equiv \log \frac{p(x,y)}{p(x)p(y)}$$

frequency

What is the relation between terms?
Hearst Patterns

- Hearst, Marti A. "Automatic acquisition of hyponyms from large text corpora." Proceedings of the 14th conference on Computational linguistics-Volume 2. Association for Computational Linguistics, 1992.
- What does Gelidium mean?
- “Αγαρ ισ α συβστανχε πρεπαρεδ φρομ α μιξτυρε οφ red algae, such as **Gelidium**, φορ λαβορατορψ ορ ινδυστριαλ υσε”
- “Agar is a substance prepared from a mixture of red algae, such as **Gelidium**, for laboratory or industrial use”

What is the relation between terms?
Hearst patterns

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...

What is the relation between terms?

Manually created patterns

- Pros
  - Tend to be high-precision
  - Can be adapted to specific domains
- Cons
  - Tend to be low-recall
  - A lot of work to define all possible patterns
  - Many different patterns are needed for every relation

Recap

- Relation extraction mainly consists of detecting binary relations between entities
- Rule-based and supervised machine learning methods can be integrated in a bootstrapping method starting from seed data and obtaining more examples
- Open versus closed information extracting, where the former uses clustering and the latter uses a schema
- Schemas can play an important role to define and restrict relations but also combine evidence
- Distant supervision exploits existing data to boost machine learning
- Performance varies depending on various factors, e.g. training data quality and volume, type of relation, difference between train and test especially in time (!)
- Ontology learning boils down to term detection and taxonomic relation detection between terms
- Statistical methods and pattern-based methods are used: the former are more robust and higher recall, the latter have more precision and lower recall