# Introduction to Information Extraction

Dr. Elena Demidova
L3S Research Center

3rd KEYSTONE Summer School
Vienna, Austria
August 22, 2017

# Motivation

Unstructured data, i.e. text, is written for humans, not machines.

Information Extraction enables machines to automatically identify information nuggets such as named entities, time expressions, relations and events in text and interlink these information nuggets with structured background knowledge.

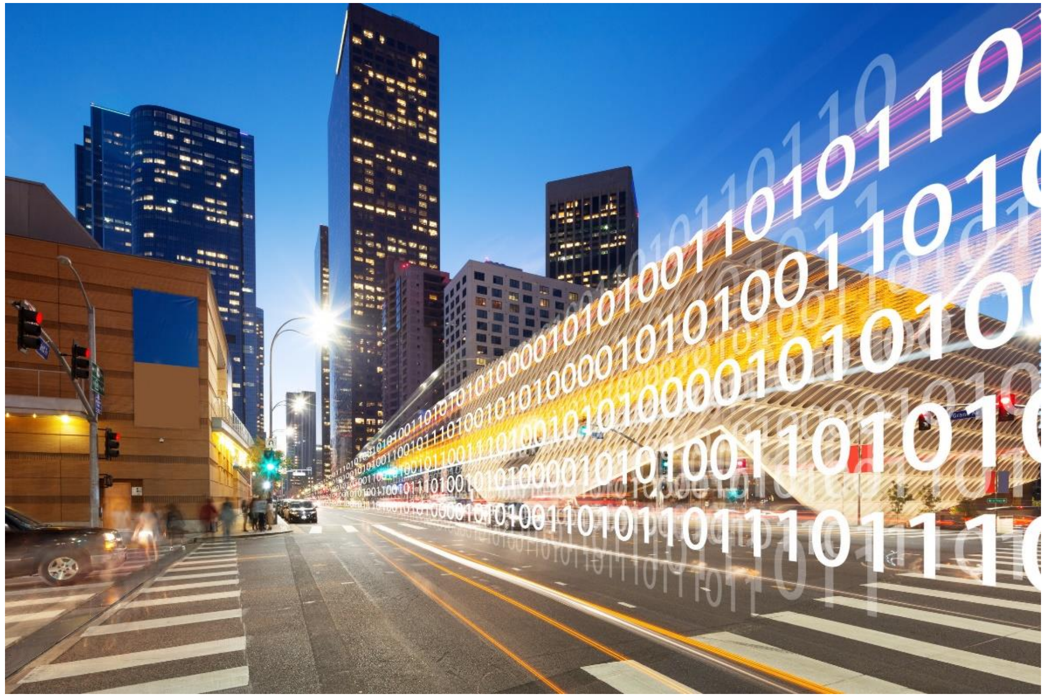
Extracted information can then be used e.g. to categorize and cluster text, enable faceted exploration, extract semantics, populate knowledge bases, correlate extracted data with other sources, e.g. across languages etc.

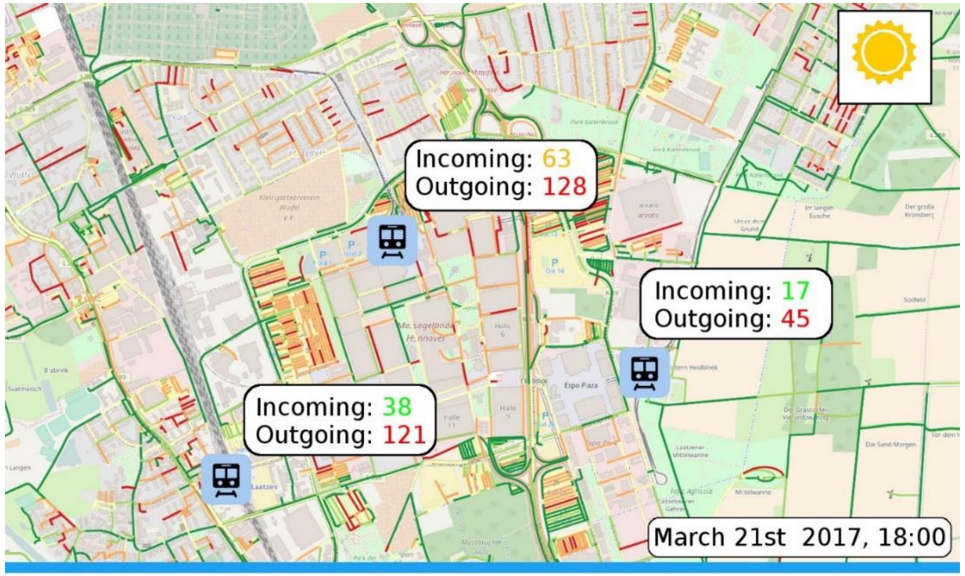
“Learn to read better than yesterday.” NELL project

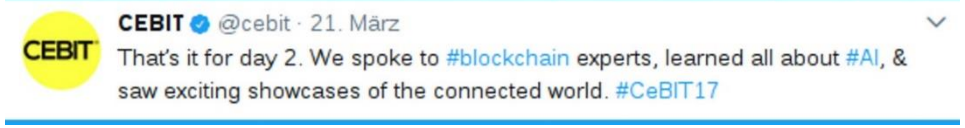
http://rtw.ml.cmu.edu/rtw/overview

# Event-Centric Data Integration

Data4UrbanMobility







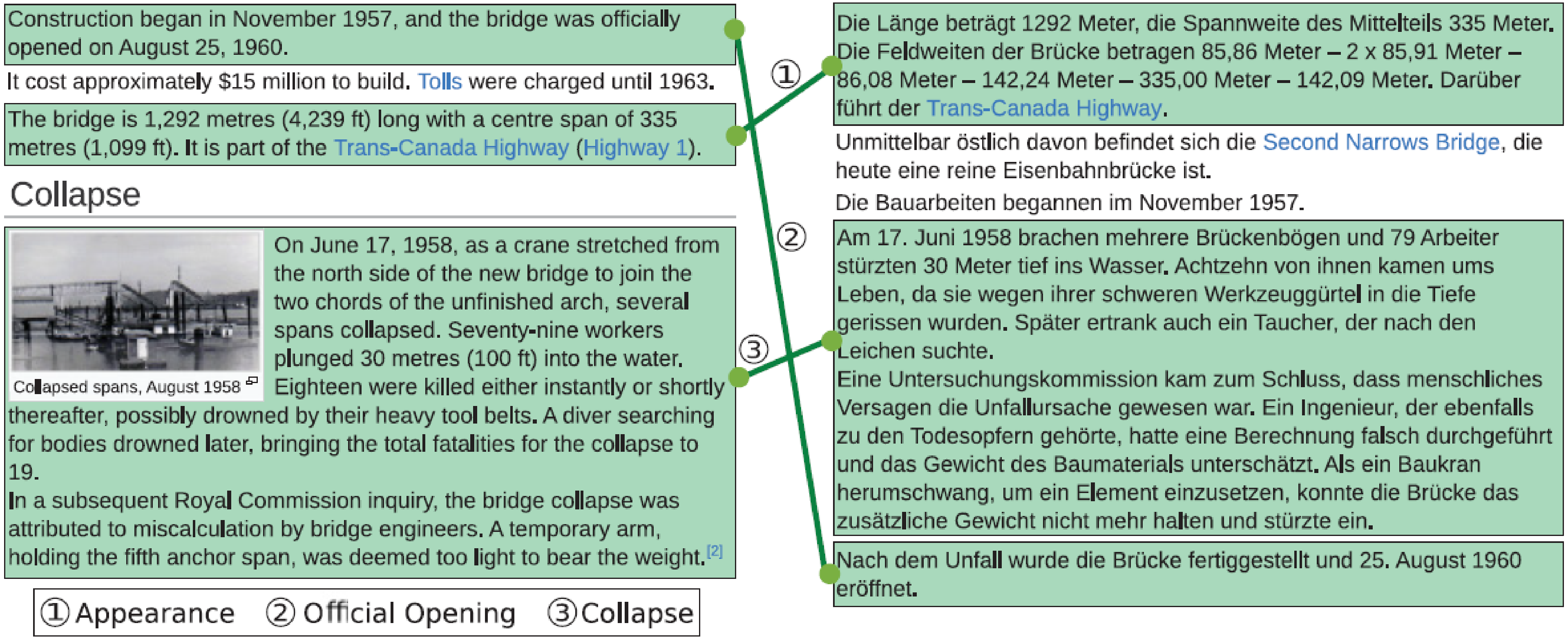
**CEBIT** @cebit · 21. März

That's it for day 2. We spoke to #blockchain experts, learned all about #AI, & saw exciting showcases of the connected world. #CeBIT17

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

data4urbanmobility.l3s.uni-hannover.de

# Multilingual Text Alignment



Construction began in November 1957, and the bridge was officially opened on August 25, 1960.

It cost approximately $15 million to build. Tolls were charged until 1963.

The bridge is 1,292 metres (4,239 ft) long with a centre span of 335 metres (1,099 ft). It is part of the Trans-Canada Highway (Highway 1).

## Collapse

Collapsed spans, August 1958

On June 17, 1958, as a crane stretched from the north side of the new bridge to join the two chords of the unfinished arch, several spans collapsed. Seventy-nine workers plunged 30 metres (100 ft) into the water. Eighteen were killed either instantly or shortly thereafter, possibly drowned by their heavy tool belts. A diver searching for bodies drowned later, bringing the total fatalities for the collapse to 19.

In a subsequent Royal Commission inquiry, the bridge collapse was attributed to miscalculation by bridge engineers. A temporary arm, holding the fifth anchor span, was deemed too light to bear the weight.[2]

Die Länge beträgt 1292 Meter, die Spannweite des Mittelteils 335 Meter. Die Feldweiten der Brücke betragen 85,86 Meter – 2 x 85,91 Meter – 86,08 Meter – 142,24 Meter – 335,00 Meter – 142,09 Meter. Darüber führt der Trans-Canada Highway.

Unmittelbar östlich davon befindet sich die Second Narrows Bridge, die heute eine reine Eisenbahnbrücke ist.

Die Bauarbeiten begannen im November 1957.

Am 17. Juni 1958 brachen mehrere Brückenbögen und 79 Arbeiter stürzten 30 Meter tief ins Wasser. Achtzehn von ihnen kamen ums Leben, da sie wegen ihrer schweren Werkzeuggürtel in die Tiefe gerissen wurden. Später ertrank auch ein Taucher, der nach den Leichen suchte.

Eine Untersuchungskommission kam zum Schluss, dass menschliches Versagen die Unfallursache gewesen war. Ein Ingenieur, der ebenfalls zu den Todesopfern gehörte, hatte eine Berechnung falsch durchgeführt und das Gewicht des Baumaterials unterschätzt. Als ein Baukran herumschwang, um ein Element einzusetzen, konnte die Brücke das zusätzliche Gewicht nicht mehr halten und stürzte ein.

Nach dem Unfall wurde die Brücke fertiggestellt und 25. August 1960 eröffnet.

① Appearance ② Official Opening ③ Collapse

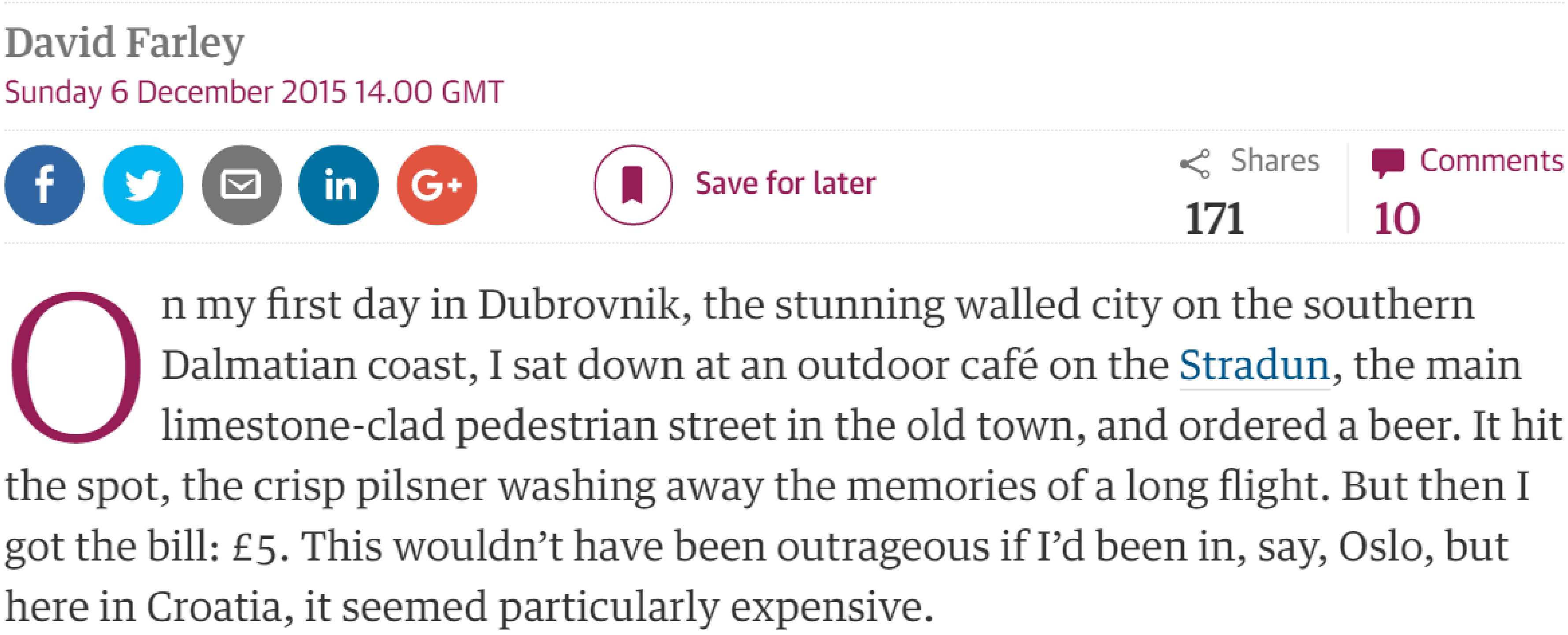
Simon Gottschalk and Elena Demidova. 2017. MultiWiki: Interlingual Text Passage Alignment in Wikipedia. *ACM Trans. Web* 11, 1, Article 6 (April 2017), 30 pages.

# Goals of the Tutorial

- Provide an overview of the methods of Information Extraction, in particular for:
  - Named Entity Extraction
  - Named Entity Linking
  - Temporal Extraction
  - Relation Extraction

- Understand how different methods of Information Extraction work
  - Rule-based approaches
  - Machine learning approaches
  - Different supervision models for machine learning

# (Semi-)structured Information on the Web



**David Farley**

Sunday 6 December 2015 14.00 GMT

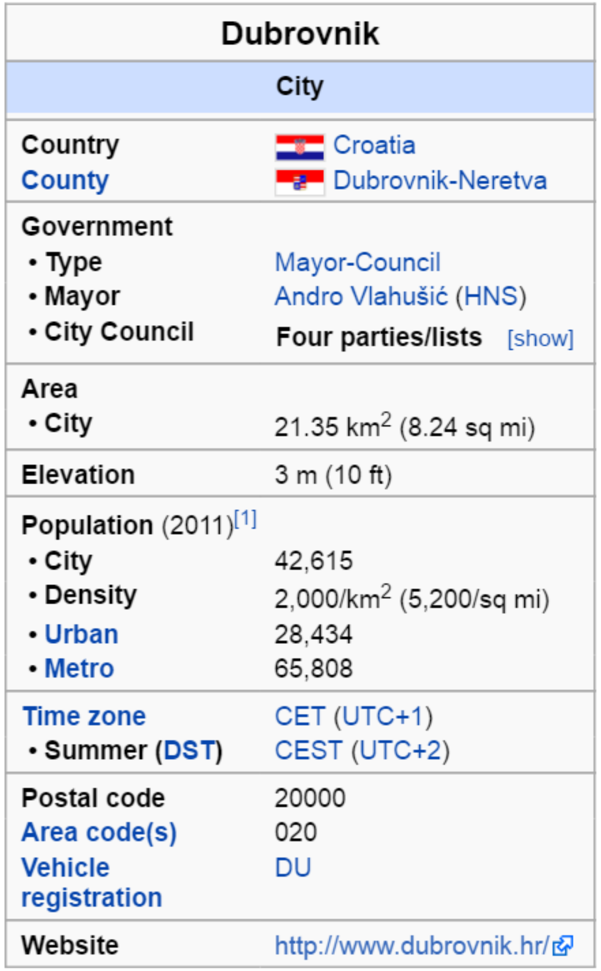
Save for later

Shares
171

Comments
10

On my first day in Dubrovnik, the stunning walled city on the southern Dalmatian coast, I sat down at an outdoor café on the Stradun, the main limestone-clad pedestrian street in the old town, and ordered a beer. It hit the spot, the crisp pilsner washing away the memories of a long flight. But then I got the bill: £5. This wouldn't have been outrageous if I'd been in, say, Oslo, but here in Croatia, it seemed particularly expensive.

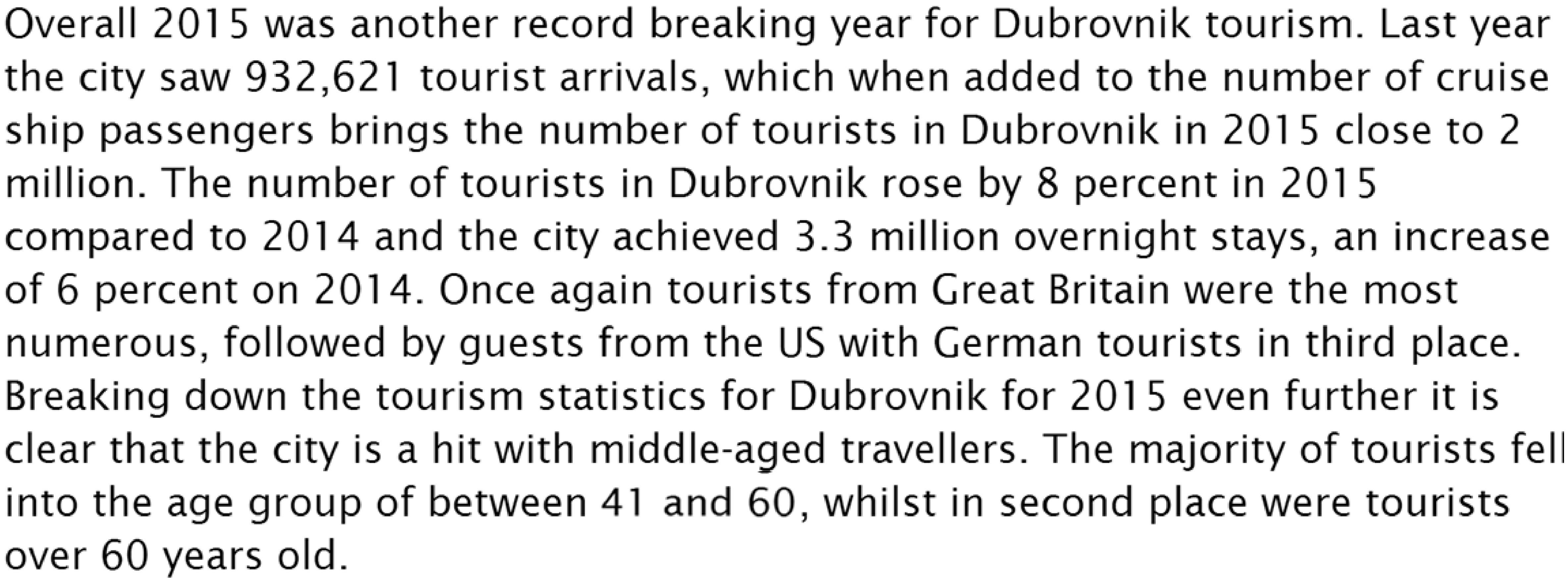
Source: https://www.theguardian.com/travel/2015/dec/06/bar-tour-dubrovnik-croatia-holiday



| Dubrovnik | |
|---|---|
| City | |
| Country | Croatia |
| County | Dubrovnik-Neretva |
| Government | |
| • Type | Mayor-Council |
| • Mayor | Andro Vlahušić (HNS) |
| • City Council | Four parties/lists [show] |
| Area | |
| • City | 21.35 km$^2$ (8.24 sq mi) |
| Elevation | 3 m (10 ft) |
| Population (2011)[1] | |
| • City | 42,615 |
| • Density | 2,000/km$^2$ (5,200/sq mi) |
| • Urban | 28,434 |
| • Metro | 65,808 |
| Time zone | CET (UTC+1) |
| • Summer (DST) | CEST (UTC+2) |
| Postal code | 20000 |
| Area code(s) | 020 |
| Vehicle registration | DU |
| Website | http://www.dubrovnik.hr/ |

Source:
https://en.wikipedia.org/wiki/Dubrovnik

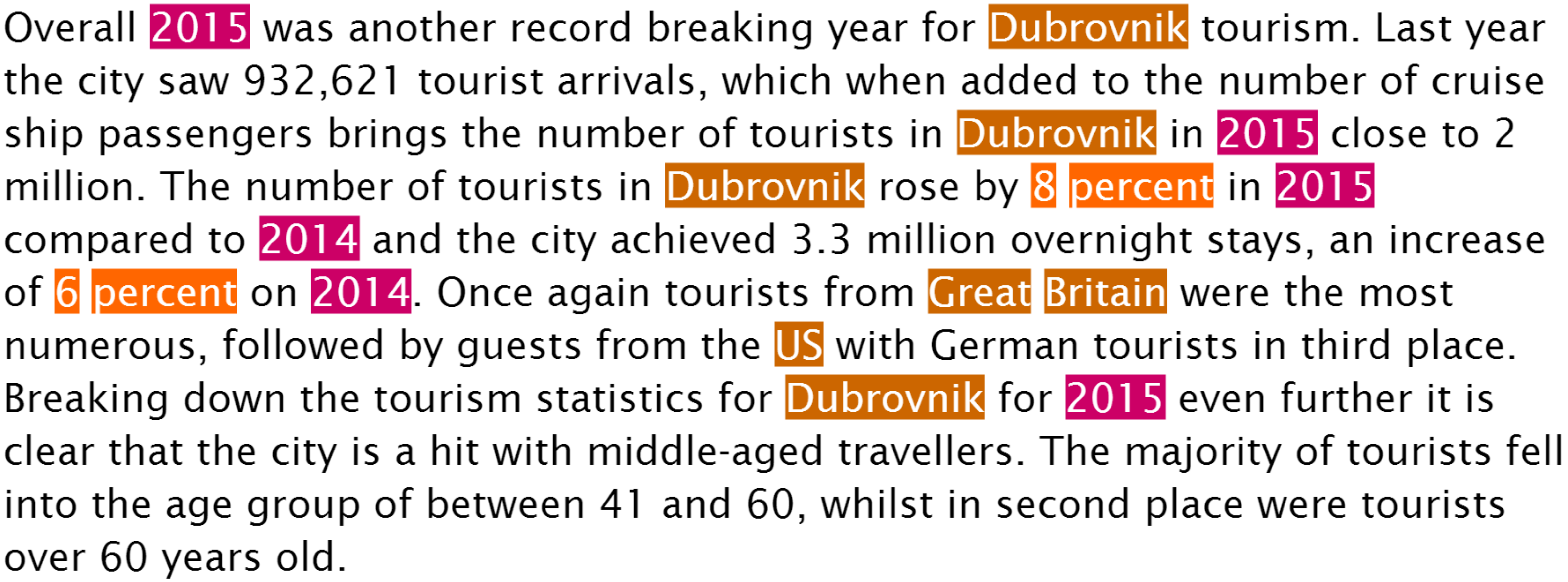
# Data “Hidden” in the Text



Overall 2015 was another record breaking year for Dubrovnik tourism. Last year the city saw 932,621 tourist arrivals, which when added to the number of cruise ship passengers brings the number of tourists in Dubrovnik in 2015 close to 2 million. The number of tourists in Dubrovnik rose by 8 percent in 2015 compared to 2014 and the city achieved 3.3 million overnight stays, an increase of 6 percent on 2014. Once again tourists from Great Britain were the most numerous, followed by guests from the US with German tourists in third place. Breaking down the tourism statistics for Dubrovnik for 2015 even further it is clear that the city is a hit with middle-aged travellers. The majority of tourists fell into the age group of between 41 and 60, whilst in second place were tourists over 60 years old.

Source: http://dubrovacki.hr/clanak/81000/2015-tourism-figures-for-dubrovnik

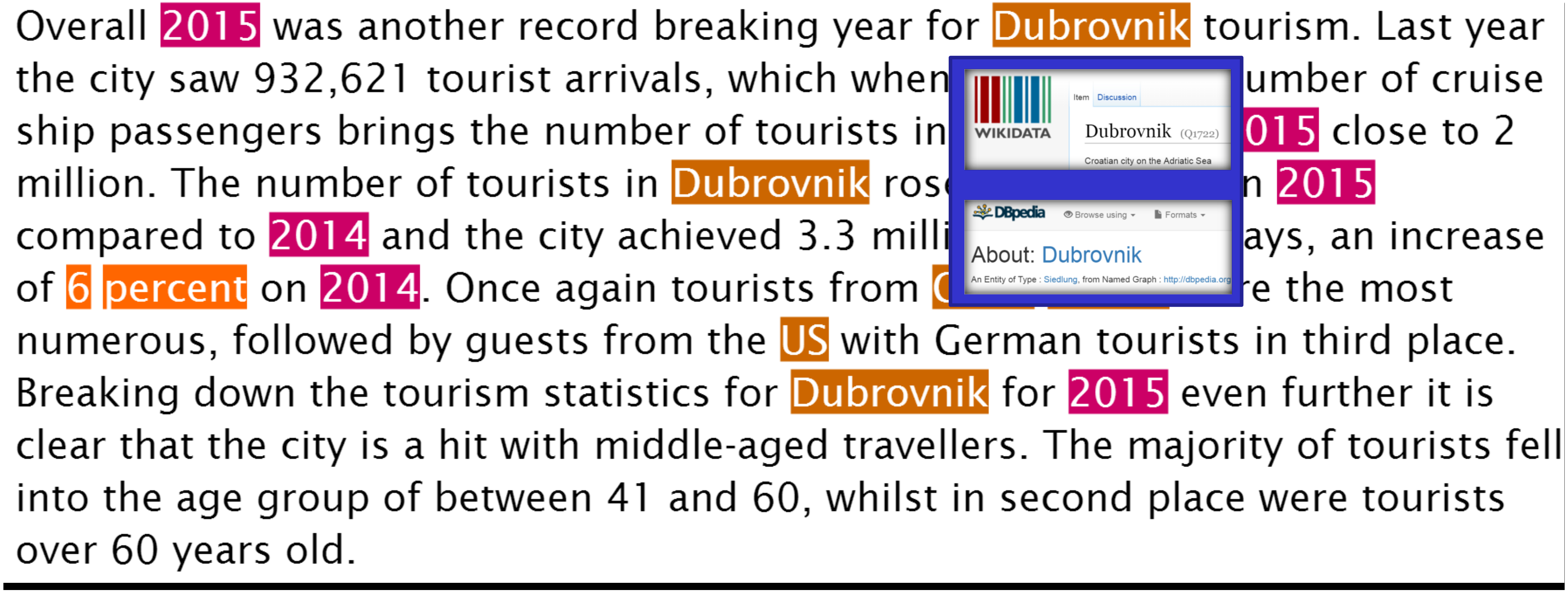
# Named Entities & Temporal Expressions



Overall 2015 was another record breaking year for Dubrovnik tourism. Last year the city saw 932,621 tourist arrivals, which when added to the number of cruise ship passengers brings the number of tourists in Dubrovnik in 2015 close to 2 million. The number of tourists in Dubrovnik rose by 8 percent in 2015 compared to 2014 and the city achieved 3.3 million overnight stays, an increase of 6 percent on 2014. Once again tourists from Great Britain were the most numerous, followed by guests from the US with German tourists in third place. Breaking down the tourism statistics for Dubrovnik for 2015 even further it is clear that the city is a hit with middle-aged travellers. The majority of tourists fell into the age group of between 41 and 60, whilst in second place were tourists over 60 years old.

Source: http://dubrovacki.hr/clanak/81000/2015-tourism-figures-for-dubrovnik

# Entity Linking



Overall 2015 was another record breaking year for Dubrovnik tourism. Last year
the city saw 932,621 tourist arrivals, which when … umber of cruise
ship passengers brings the number of tourists in … 015 close to 2
million. The number of tourists in Dubrovnik rose … n 2015
compared to 2014 and the city achieved 3.3 milli … ays, an increase
of 6 percent on 2014. Once again tourists from C … re the most
numerous, followed by guests from the US with German tourists in third place.
Breaking down the tourism statistics for Dubrovnik for 2015 even further it is
clear that the city is a hit with middle-aged travellers. The majority of tourists fell
into the age group of between 41 and 60, whilst in second place were tourists
over 60 years old.

WIKIDATA
Item Discussion
Dubrovnik (Q1722)
Croatian city on the Adriatic Sea

DBpedia
Browse using
Formats
About: Dubrovnik
An Entity of Type : Siedlung, from Named Graph : http://dbpedia.org

Sources: http://dubrovacki.hr/clanak/81000/2015-tourism-figures-for-Dubrovnik, https://www.wikidata.org/wiki/Q1722, http://dbpedia.org/page/Dubrovnik

# Information Extraction (IE)

- IE is the task of identification of structured information in text. IE includes:
  - **Named Entity extraction and disambiguation**
    - Dubrovnik is a city. Dubrovnik ->http://dbpedia.org/resource/Dubrovnik
  - **Extraction of temporal expressions**
    - 10th July, 25th August 2016.
  - **Extraction of relations between Named Entities**
    - Dubrovnik is located in the region of Dalmatia.
  - **Event extraction**
    - One of Croatia's most famous events, the Dubrovnik Summer Festival, took place from 10th July to 25th August 2016.

# Named Entity Extraction: Terminology

**Named Entities**: Proper nouns or phrases, which refer to real-world objects (entities).

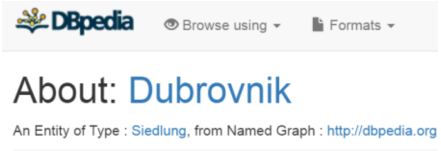
**Named Entity Extraction (Recognition, Identification)**: Detecting boundaries of named entities (NEs) in unstructured text.

**Named Entity Classification**: Automatically assigning pre-defined classes to NEs, such as PERSON, LOCATION, ORGANISATION, etc.

**Named Entity Linking / Disambiguation**: Linking NEs to entries in a knowledge base (e.g. DBpedia, Wikidata, etc.):

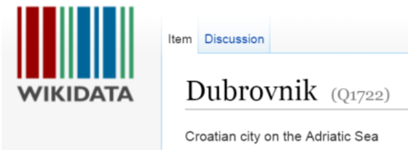
Dubrovnik -> [http://dbpedia.org/resource/Dubrovnik](http://dbpedia.org/resource/Dubrovnik)

-> [https://www.wikidata.org/wiki/Q1722](https://www.wikidata.org/wiki/Q1722)



About: Dubrovnik

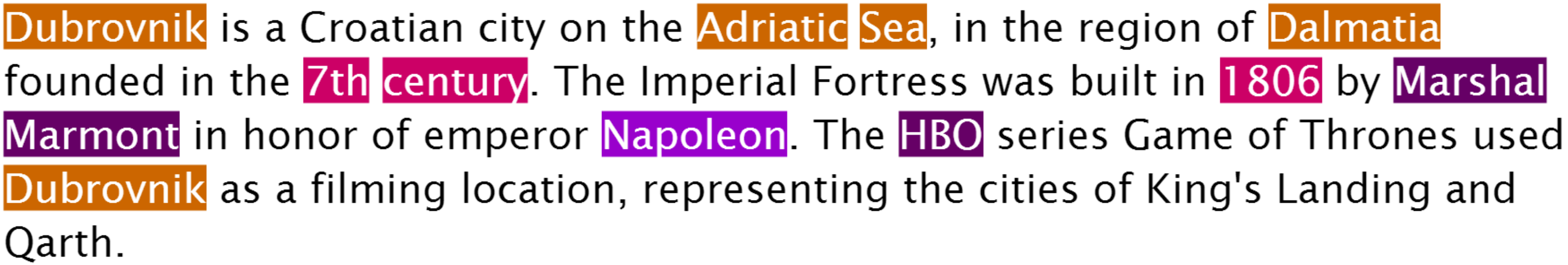
An Entity of Type : Siedlung, from Named Graph : http://dbpedia.org



Dubrovnik (Q1722)

Croatian city on the Adriatic Sea

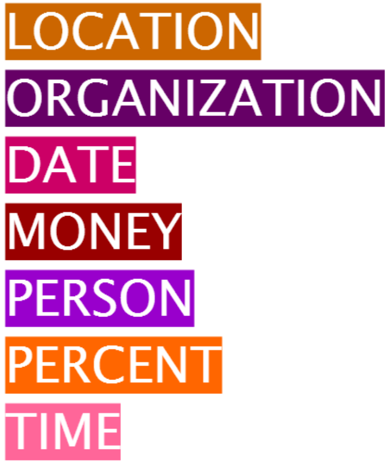
# Named Entity Extraction: Examples



Dubrovnik is a Croatian city on the Adriatic Sea, in the region of Dalmatia founded in the 7th century. The Imperial Fortress was built in 1806 by Marshal Marmont in honor of emperor Napoleon. The HBO series Game of Thrones used Dubrovnik as a filming location, representing the cities of King's Landing and Qarth.

Extraction by Stanford Named Entity Tagger

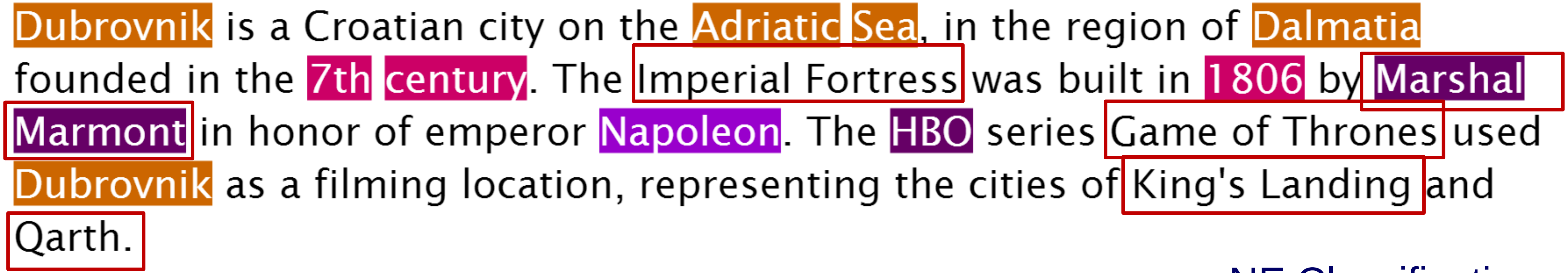
NE Classification



LOCATION
ORGANIZATION
DATE
MONEY
PERSON
PERCENT
TIME

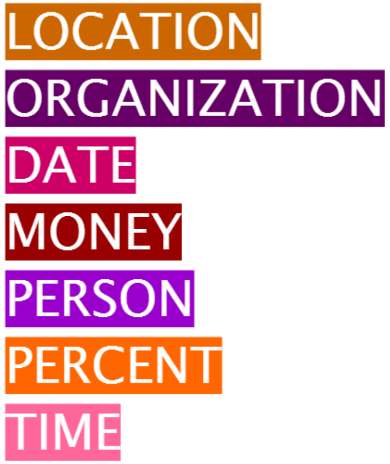
**Any issues?**

# Named Entity Extraction: Examples



Dubrovnik is a Croatian city on the Adriatic Sea, in the region of Dalmatia founded in the 7th century. The Imperial Fortress was built in 1806 by Marshal Marmont in honor of emperor Napoleon. The HBO series Game of Thrones used Dubrovnik as a filming location, representing the cities of King's Landing and Qarth.

## NE Classification



LOCATION
ORGANIZATION
DATE
MONEY
PERSON
PERCENT
TIME

**Problems:**

- Unknown entities
- Unknown entity types
- Ambiguities / wrong types

# Named Entity Extraction: Methods

- **Rule-based approaches**: Using hand-coded extraction rules
- **Machine learning based approaches**
  - Supervised learning (domain specific): Manually annotate the text, train a model
  - Unsupervised learning (Web-scale NER): Extract language patterns, cluster similar ones
  - Semi-supervised learning: Start with a small number of language patterns, iteratively learn more (bootstrapping)
- **Methods based on existing resources**
  - Gazetteer-based method: Use existing list of named entities
  - Using Web resources and KBs: Wikipedia, DBpedia, Web n-gram corpora, etc.
- **Combinations of the methods above**

# NERC: Choice of Machine Learning Algorithms

see: http://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

# NE Extraction Pipeline

**Pre-processing of text**

- Text extraction (mark up removal), sentence splitting, tokenization (identification of individual terms)

**Linguistic pre-processing of tokens**

- Lemmatisation (lexicon) or stemming (algorithms):
    - reduce inflectional forms of a word to a common base form
- Part of speech (POS) tagging

**Chunking (shallow parsing), parsing (parse tree)**

- Noun phrases, grammatical relations

**Semantic and discourse analysis, anaphora resolution (co-references)**

- What actions are being described? What roles entities play in this actions? How does they relate to other entities and actions in other sentences?

# NE Extraction Pipeline

- Sentence splitting: *Dubrovnik is located in the region of Dalmatia.*

# NE Extraction Pipeline

- Sentence splitting: *Dubrovnik is located in the region of Dalmatia.*
- Tokenization: Dubrovnik is located in the region of Dalmatia.

杜布羅夫尼克位於達爾馬提亞地區。

# NE Extraction Pipeline

- Sentence splitting: *Dubrovnik is located in the region of Dalmatia.*
- Tokenization: Dubrovnik | is | located | in | the | region | of | Dalmatia.

Lemmatisation or stemming (reduce inflectional forms of a word to a common base form):

  - E.g. “located” -> “locate”

# NE Extraction Pipeline

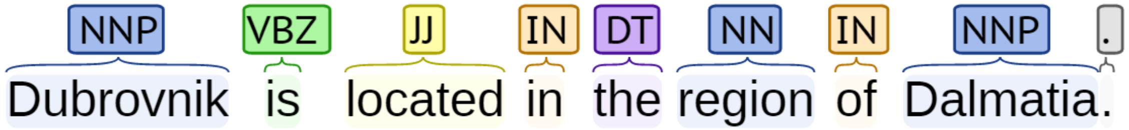
- Sentence splitting: *Dubrovnik is located in the region of Dalmatia.*
- Tokenization:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Dubrovnik | is | located | in | the | region | of | Dalmatia. |

Lemmatisation or stemming:

  - E.g. “located” -> “locate”

POS tagging: Nouns, adjectives and verbs



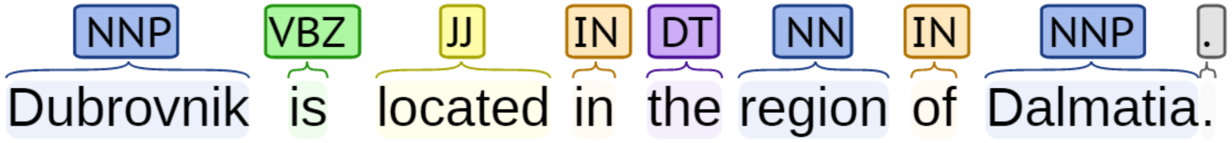
# Morphology: Penn Treebank POS tags

| Number | Tag | Description |
|---|---|---|
| 1. | CC | Coordinating conjunction |
| 2. | CD | Cardinal number |
| 3. | DT | Determiner |
| 4. | EX | Existential *there* |
| 5. | FW | Foreign word |
| 6. | IN | Preposition or subordinating conjunction |
| 7. | JJ | Adjective |
| 8. | JJR | Adjective, comparative |
| 9. | JJS | Adjective, superlative |
| 10. | LS | List item marker |
| 11. | MD | Modal |
| 12. | NN | Noun, singular or mass |
| 13. | NNS | Noun, plural |
| 14. | NNP | Proper noun, singular |
| 15. | NNPS | Proper noun, plural |
| 16. | PDT | Predeterminer |
| 17. | POS | Possessive ending |
| 18. | PRP | Personal pronoun |
| 19. | PRP$ | Possessive pronoun |
| 20. | RB | Adverb |
| 21. | RBR | Adverb, comparative |
| 22. | RBS | Adverb, superlative |
| 23. | RP | Particle |
| 24. | SYM | Symbol |
| 25. | TO | *to* |
| 26. | UH | Interjection |
| 27. | VB | Verb, base form |
| 28. | VBD | Verb, past tense |
| 29. | VBG | Verb, gerund or present participle |
| 30. | VBN | Verb, past participle |
| 31. | VBP | Verb, non-3rd person singular present |
| 32. | VBZ | Verb, 3rd person singular present |
| 33. | WDT | |
| 34. | WP | |
| 35. | WP$ | |
| 36. | WRB | |

**Adjectives (all start with *J*)**

**Nouns (all start with *N*)**

**Verbs (all start with *V*)**



Source: Isabelle Augenstein, Information Extraction with Linked Data Tutorial, ESWC Summer School 2015

# NE Extraction Pipeline

- Sentence splitting: *Dubrovnik is located in the region of Dalmatia.*
- Tokenization: Dubrovnik | is | located | in | the | region | of | Dalmatia.

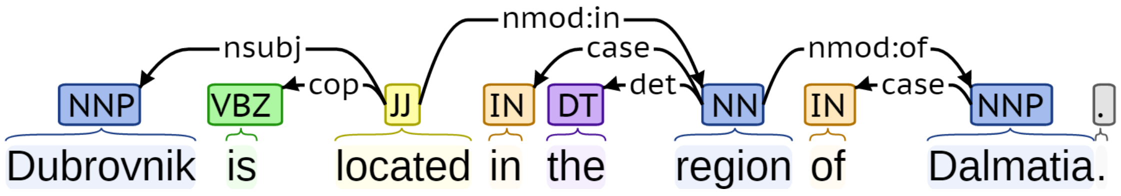
Lemmatisation or stemming:

  - E.g. “located” -> “locate”

POS tagging: Nouns, adjectives and verbs

Chunking, parsing:

  - “Dubrovnik is located”, “region of Dalmatia”



Noun phrases, grammatical relations

nsubj: nominal subject - a noun phrase, the syntactic subject of a clause.

# NE Extraction Pipeline

- Sentence splitting: *Dubrovnik is located in the region of Dalmatia.*
- Tokenization: Dubrovnik | is | located | in | the | region | of | Dalmatia.

Lemmatisation or stemming:

- E.g. “located” -> “locate”

POS tagging: Nouns, adjectives and verbs

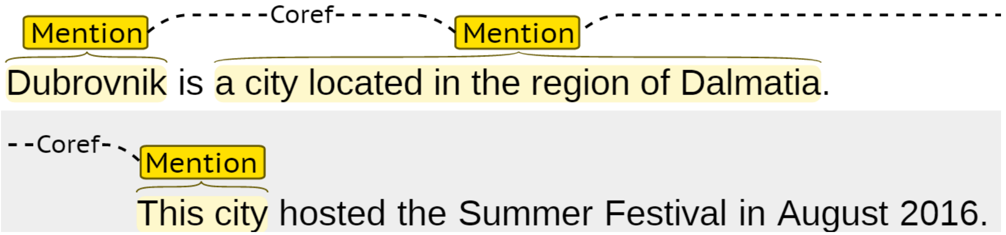
Chunking, parsing:

- “Dubrovnik is located”, “region of Dalmatia”

Co-reference resolution:

- “Dubrovnik” and “This city”.

(Examples: Stanford NER)



Dubrovnik is a city located in the region of Dalmatia.

This city hosted the Summer Festival in August 2016.

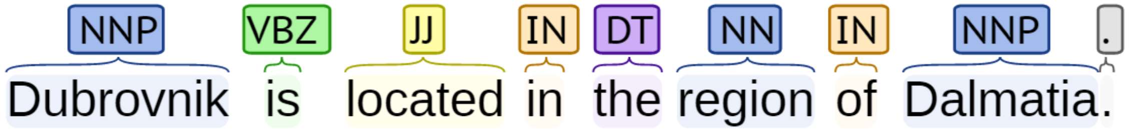
# NE Extraction Pipeline

- Sentence splitting: *Dubrovnik is located in the region of Dalmatia.*
- Tokenization: Dubrovnik | is | located | in | the | region | of | Dalmatia.

Lemmatisation or stemming:

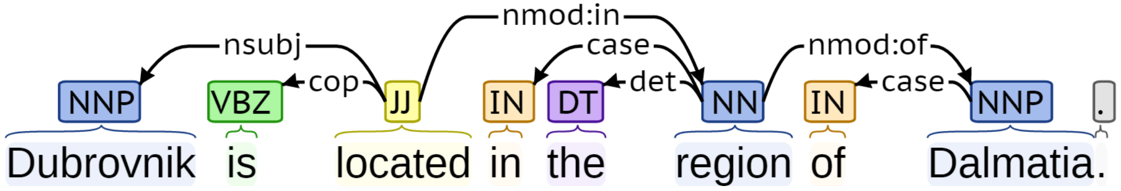
- E.g. “located” -> “locate”

POS tagging: Nouns, adjectives and verbs



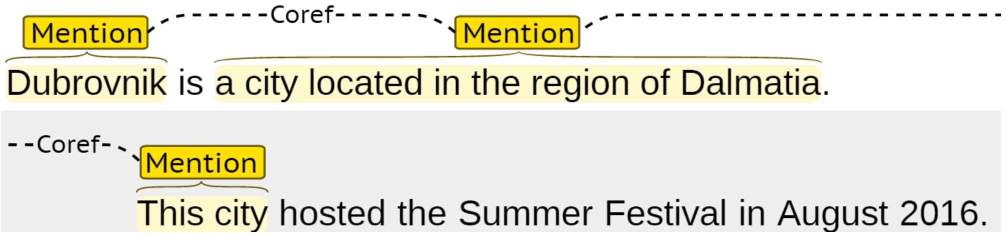
Chunking, parsing:

- “Dubrovnik is located”, “region of Dalmatia”



Co-reference resolution:

- “Dubrovnik” and “This city”.


Mention
Coref
Mention
Dubrovnik is a city located in the region of Dalmatia.
Coref
Mention
This city hosted the Summer Festival in August 2016.

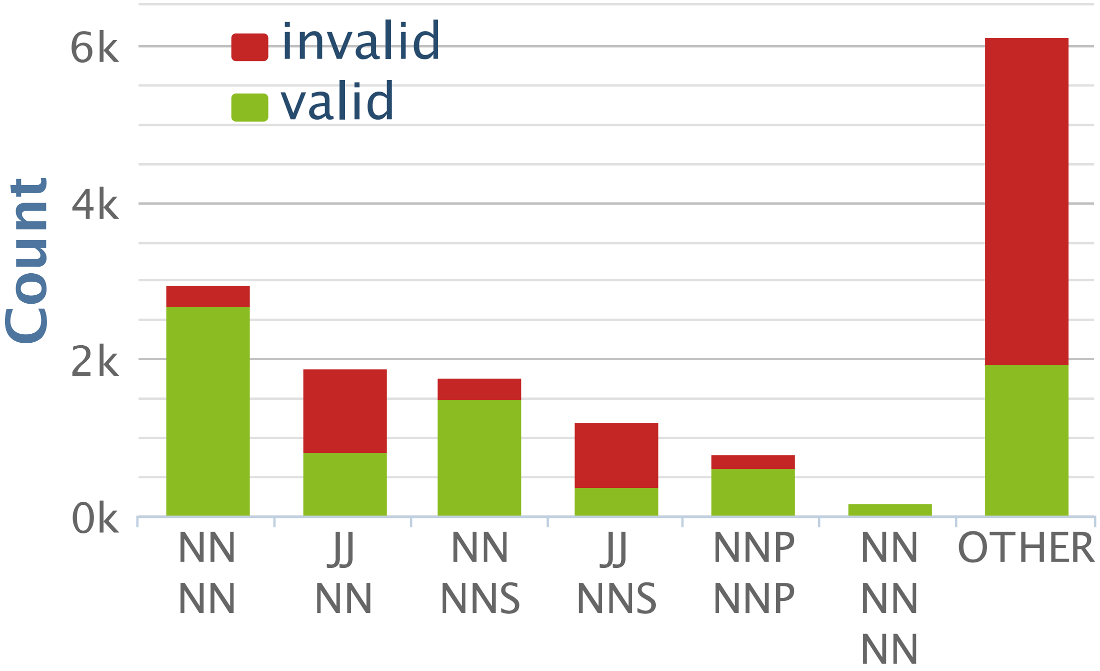
(Examples: Stanford NER)

# Named Entity Extraction: Features

- Words:
  - Words in window before and after mention
  - Sequences (n-grams), frequencies
  - Bags of words, word2vec
- Morphology:
  - Capitalization: is upper case (*China*), all upper case (*IBM*), mixed case (*eBay*)
  - Symbols: contains $, £, €, roman symbols (*IV*), ..
  - Contained special characters: period (*google.com*), apostrophe (*Mandy’s*), hyphen (*speed-o-meter*), ampersand (*Fisher & Sons*)
  - Stem or lemma (*cats->cat*), prefix (*disadvantages->dis*), suffix (*cats->s*), interfix (*speed-o-meter->o*)

# Named Entity Extraction: Features

- POS tags, POS tag patterns
  - NN and NNS singular and plural nouns
  - NNP proper nouns
  - JJ adjectives
- Near n-Gram punctuation
- Other n-gram related features

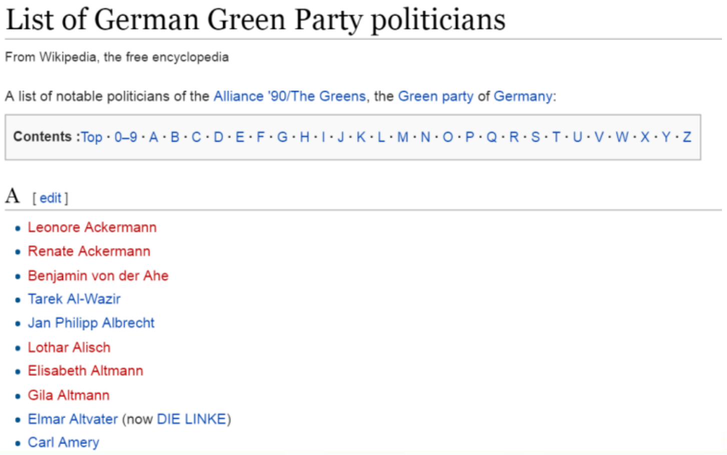


Top 6 most frequent part-of-speech tag patterns of the SIGIR collection.

(Prokofyev 2014)

# Named Entity Extraction: Features

- Gazetteers
  - Using regular expressions patterns and search engines (e.g. “*Popular artists such as * *”)
  - Retrieved from knowledge bases
    - General: Wikipedia, DBpedia, Wikidata, (Freebase)
    - Domain-specific: DBLP, Physics Concepts, etc.
  - Retrieved from the Web tables and lists



List of German Green Party politicians

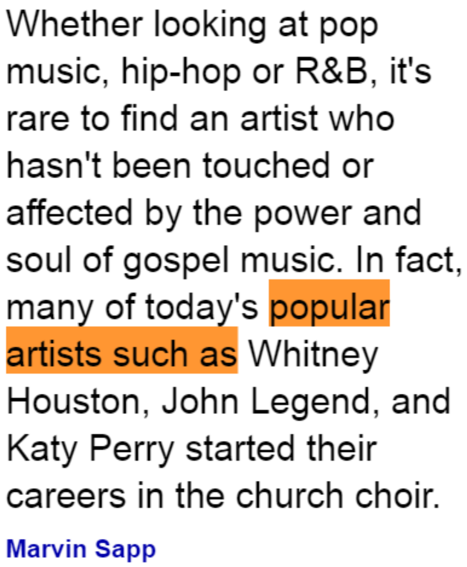
From Wikipedia, the free encyclopedia

A list of notable politicians of the Alliance '90/The Greens, the Green party of Germany:

**Contents** :Top · 0–9 · A · B · C · D · E · F · G · H · I · J · K · L · M · N · O · P · Q · R · S · T · U · V · W · X · Y · Z

A [ edit ]

- Leonore Ackermann
- Renate Ackermann
- Benjamin von der Ahe
- Tarek Al-Wazir
- Jan Philipp Albrecht
- Lothar Alisch
- Elisabeth Altmann
- Gila Altmann
- Elmar Altvater (now DIE LINKE)
- Carl Amery



Whether looking at pop music, hip-hop or R&B, it's rare to find an artist who hasn't been touched or affected by the power and soul of gospel music. In fact, many of today's popular artists such as Whitney Houston, John Legend, and Katy Perry started their careers in the church choir.

**Marvin Sapp**

Sources: https://en.wikipedia.org/wiki/List_of_German_Green_Party_politicians
http://www.brainyquote.com/quotes/keywords/artist.html

# Named Entity Extraction: Evaluation Measures

- Precision
  - Proportion of correctly extracted NEs among all extracted NEs.
- Recall
  - Proportion of NEs found by the algorithm to all NEs in the collection.
- F-Measure
  - The weighted harmonic mean of precision and recall.

# Open Problems in NER

Extraction does not work equally well in all domains

- Specialised technical texts
- Other languages / multilingual text collections

Newly emerging / unknown entities (e.g. in the context of news events)

- Edward Snowden before the NSA scandal
- Regional entities (e.g. not widely known politicians)
- Annotating named entities in local news papers

Entity evolution (entity name or attribute changes over time)

- St. Petersburg vs. Leningrad and Petrograd
- Pope Francis vs. Jorge Mario Bergoglio

"…Bombay, also known as **Mumbai**…

-January 09, 2000 - Arts – Article NYTimes

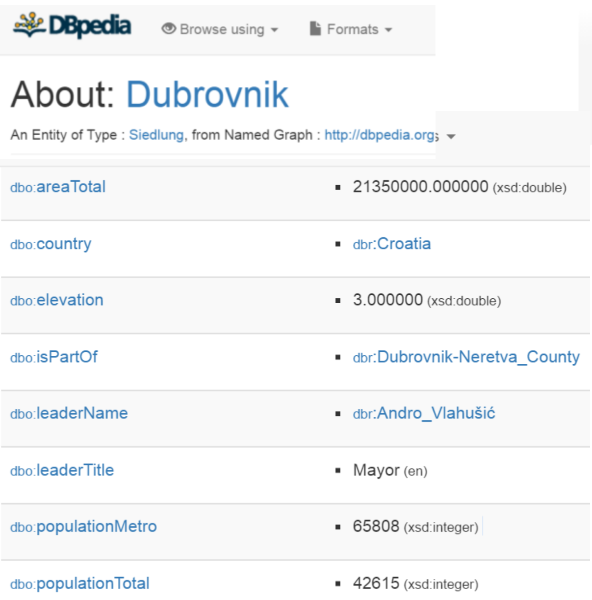
# Entity Linking

Entity Linking (EL): detecting entities and linking them to the entries of a Knowledge Base

**Dubrovnik** is located in the region of Dalmatia.

-> http://dbpedia.org/resource/Dubrovnik

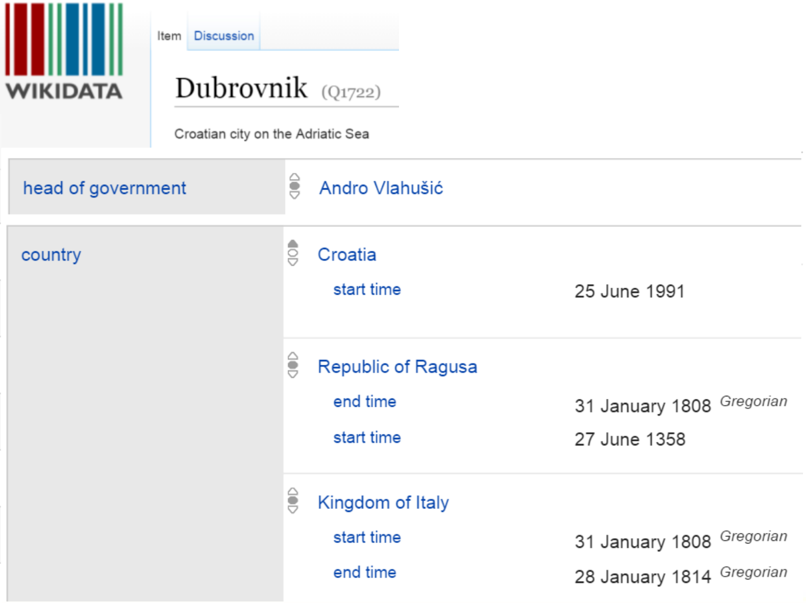
-> https://www.wikidata.org/wiki/Q1722



About: Dubrovnik

An Entity of Type : Siedlung, from Named Graph : http://dbpedia.org

| | |
|---|---|
| dbo:areaTotal | ▪ 21350000.000000 (xsd:double) |
| dbo:country | ▪ dbr:Croatia |
| dbo:elevation | ▪ 3.000000 (xsd:double) |
| dbo:isPartOf | ▪ dbr:Dubrovnik-Neretva_County |
| dbo:leaderName | ▪ dbr:Andro_Vlahušić |
| dbo:leaderTitle | ▪ Mayor (en) |
| dbo:populationMetro | ▪ 65808 (xsd:integer) |
| dbo:populationTotal | ▪ 42615 (xsd:integer) |



Dubrovnik (Q1722)

Croatian city on the Adriatic Sea

| | | | |
|---|---|---|---|
| head of government | Andro Vlahušić | | |
| country | Croatia | | |
| | | start time | 25 June 1991 |
| | Republic of Ragusa | | |
| | | end time | 31 January 1808 Gregorian |
| | | start time | 27 June 1358 |
| | Kingdom of Italy | | |
| | | start time | 31 January 1808 Gregorian |
| | | end time | 28 January 1814 Gregorian |

# Entity Linking: Motivation

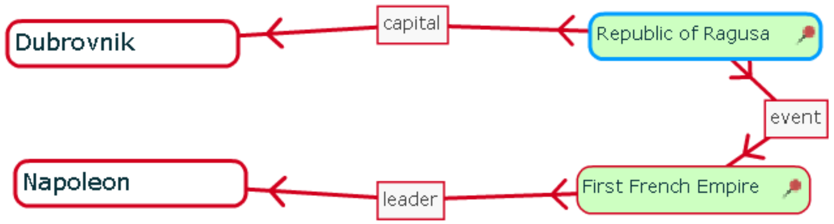
Provide additional information / facts about the entities in the text

Uncover relations between entities

**Dubrovnik**

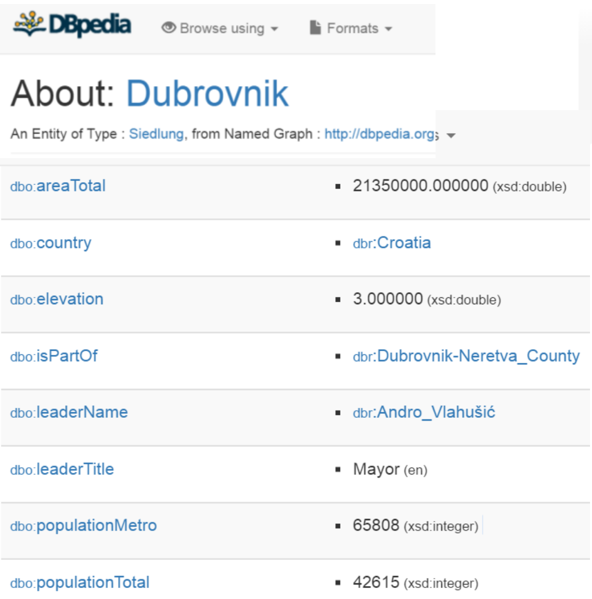
-> http://dbpedia.org/resource/Dubrovnik

-> https://www.wikidata.org/wiki/Q1722



http://www.visualdataweb.org/relfinder/relfinder.php

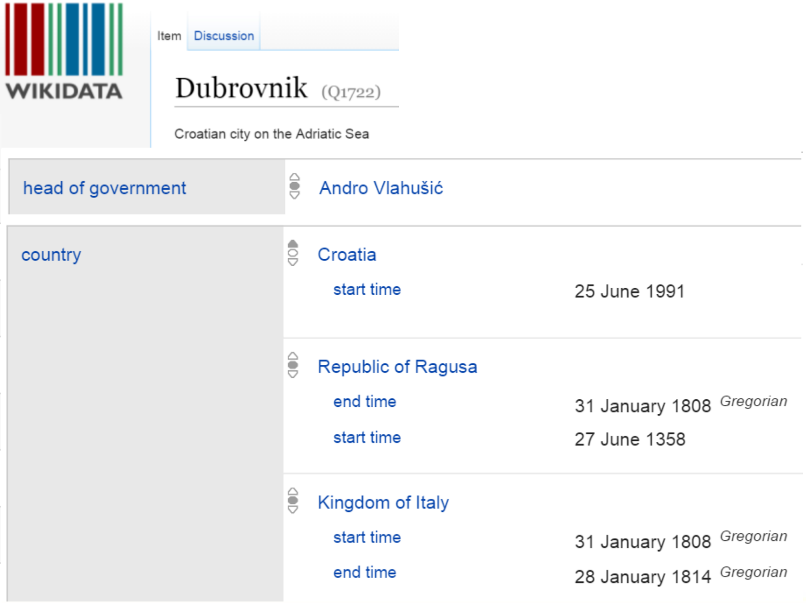
The Republic of Ragusa was a maritime republic centered on the city of Dubrovnik. It was conquered by Napoleon's French Empire in 1808.



About: Dubrovnik

An Entity of Type : Siedlung, from Named Graph : http://dbpedia.org

| | |
|---|---|
| dbo:areaTotal | 21350000.000000 (xsd:double) |
| dbo:country | dbr:Croatia |
| dbo:elevation | 3.000000 (xsd:double) |
| dbo:isPartOf | dbr:Dubrovnik-Neretva_County |
| dbo:leaderName | dbr:Andro_Vlahušić |
| dbo:leaderTitle | Mayor (en) |
| dbo:populationMetro | 65808 (xsd:integer) |
| dbo:populationTotal | 42615 (xsd:integer) |



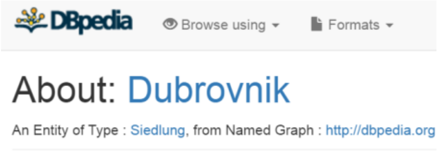
Dubrovnik (Q1722)

Croatian city on the Adriatic Sea

| | | | |
|---|---|---|---|
| head of government | Andro Vlahušić | | |
| country | Croatia | | |
| | | start time | 25 June 1991 |
| | Republic of Ragusa | | |
| | | end time | 31 January 1808 Gregorian |
| | | start time | 27 June 1358 |
| | Kingdom of Italy | | |
| | | start time | 31 January 1808 Gregorian |
| | | end time | 28 January 1814 Gregorian |

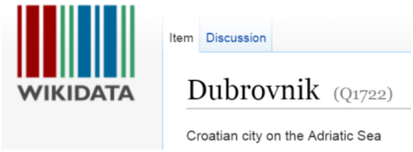
# Entity Linking: Related Problems

- **Knowledge Base population**
  - Populate a KB with named entities identified in text
    - -> lifting unstructured data into a pre-defined structure
- **Interlinking records across databases**
  - Determine records represent the same entity to be merged (referred to as object identification, data de-duplication, entity resolution, entity disambiguation and record linkage)
    - -> matching structured data instances
- **Co-reference resolution or entity resolution**
  - Clustering entity mentions either within the same document or across multiple documents together, where each cluster corresponds to a single real-world entity
    - -> matching instances extracted from unstructured data



About: Dubrovnik

An Entity of Type : Siedlung, from Named Graph : http://dbpedia.org



Dubrovnik (Q1722)

Croatian city on the Adriatic Sea

Hong-jie Dai , Chi-yang Wu , Richard Tzong-han , Tsai Wen-lian Hsu
From Entity Recognition to Entity Linking: A Survey of Advanced Entity Linking Techniques

# Entity Linking: Pipeline

**Spotting**

- Detecting all non-overlapping strings in a text that could mention an entity
- Methods: Named Entity Recognition, detecting multi-word entities, finding sequences of capitalized words, surface form dictionary

**Candidate generation**

- Finding all possible candidate entities in KB that may be referred to the spotted string
- Methods: query expansion and matching

**Candidate disambiguation**

- Selection of the most likely candidate in KB (if any)
- Methods: ranking, classification

# Entity Linking: Disambiguation Challenges

**Name variation / evolution**

- The same entity can be referred to by different names
  - Pope Francis, Franciscus, Jorge Mario Bergoglio
- Methods: Dictionary

**String ambiguity**

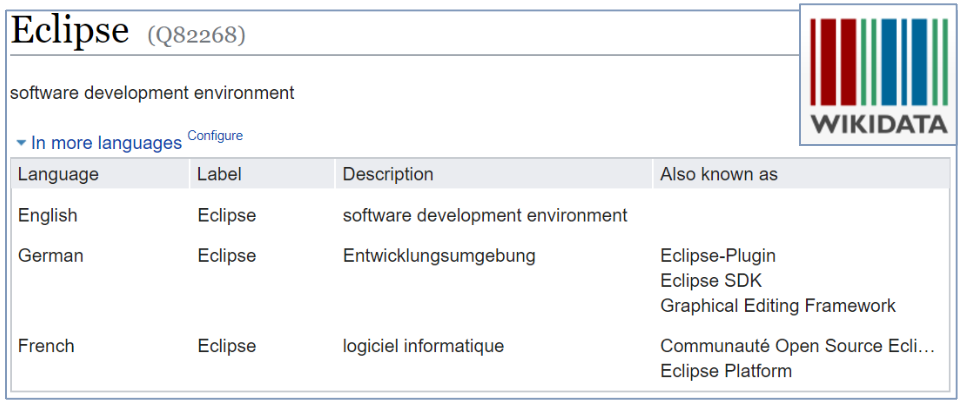
- The same name string can refer to more than one entity
- Methods: use of context
  - "*Eclipse, is a 2010 American romantic fantasy film*"
  - *"Eclipse is famous for its Java IDE*"
  - "*On August 21, 2017, North America was treated to an eclipse of the sun.*"

**Absence / KB incompleteness**

- Many mentioned entities may not appear in a KB (NIL)
- Methods: classification, thresholds

# Entity Linking: Disambiguation Features

*"Eclipse is famous for its* ***Java*** *IDE"*

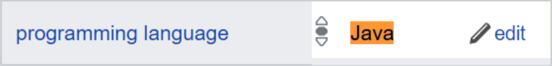


Eclipse (Q82268)

software development environment

In more languages Configure

| Language | Label | Description | Also known as |
|---|---|---|---|
| English | Eclipse | software development environment | |
| German | Eclipse | Entwicklungsumgebung | Eclipse-Plugin<br>Eclipse SDK<br>Graphical Editing Framework |
| French | Eclipse | logiciel informatique | Communauté Open Source Ecli…<br>Eclipse Platform |



programming language: Java

## Statistics

- TF-IDF (frequency and selectivity of candidates)

## Entity context similarity

- Context in the observed phrase and in the textual description in the KB
- Dependencies among entities in text and KB

## Entity type information

- Restrictions to specific types (e.g. PERSON, LOC, ORG or domain-specific)

## Link-based measures

- Popularity: hyperlinks between entities in KB (normalised inlink count or PageRank)
- Link context: anchor text (e.g. in Wikipedia)

# Entity Linking: Selected Tools

**DBpedia Spotlight (hands on session)**

- DBpedia annotations http://wiki.dbpedia.org/projects/dbpedia-spotlight

**AIDA**

- Maps mentions of ambiguous names onto canonical entities (e.g., individual people or places) registered in the YAGO2 knowledge base https://github.com/yago-naga/aida

**Illinois Wikifier**

- Disambiguating concepts and entities in a context sensitive way in Wikipedia http://cogcomp.org/page/software_view/Wikifier

**Babelfy (hands on session)**

- Babelfy is a unified, multilingual, graph-based approach to Entity Linking and Word Sense Disambiguation. http://babelfy.org/about

# Temporal Extraction

**Temporal extraction** is the extraction and normalization of temporal expressions.

- **TimeML**: specification language for temporal expressions (Pustejovsky et al. 2005)

**Types of temporal expressions in TimeML**

- Date: ‘‘August 23, 2017’’, “tomorrow”.
- Time: ‘‘11 a.m.’’, “3 in the afternoon”.
- Duration (length of an interval): ‘‘three years’’, “since yesterday”.
- Set (periodical aspect of an event): ‘‘twice a month’’.

# Temporal Extraction: Pipeline

**Pre-processing**

- Linguistic pre-processing documents with sentence, token, and POS tagging.
- Identification of the publication date (e.g. for news).

**Pattern extraction**

- Goal: decide whether a token is part of a temporal expression.
- Methods: rule-based or machine learning (as a classification problem).
- Features, patterns: terms frequently used to form temporal expressions; names of months and weekdays or numbers that may refer to a day or year; POS, context information.

**Normalization**

- Goal: assign temporal expression a value in a standard format.
- Methods: rule-based. E.g. “January” -> “01”.
- Output: type (date, time, duration, set) and value.

# Temporal Extraction

**Explicit expressions**

- Fully specified and can thus be normalized without any further knowledge
- August 22, 2017

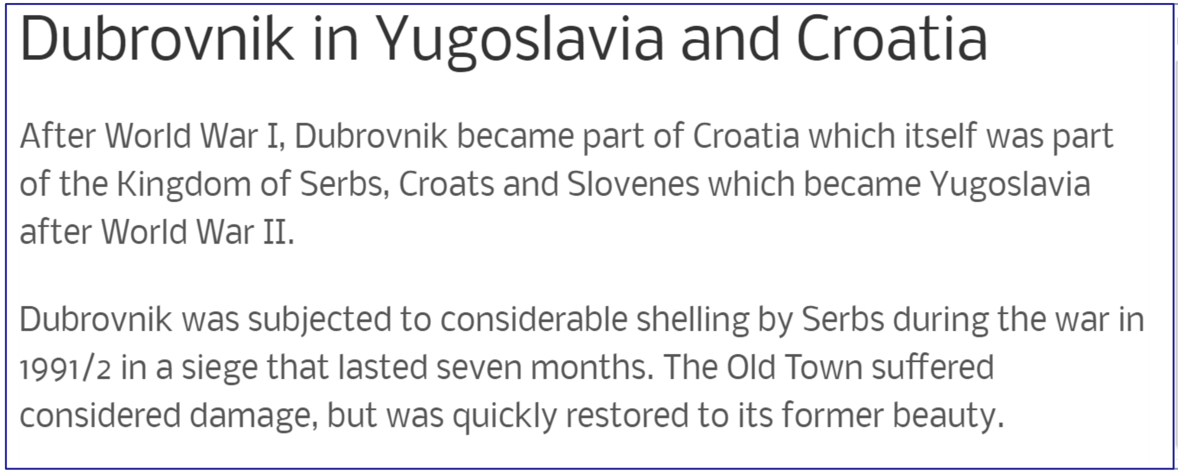
**Implicit expressions**

- Names of days and events that can directly be associated with a point or interval in time
- KEYSTONE Summer School 2017, Christmas 2017, World War II

**Relative expressions**

- Require context to normalize
- “Today”, “the following year”

# Temporal Extraction: Examples



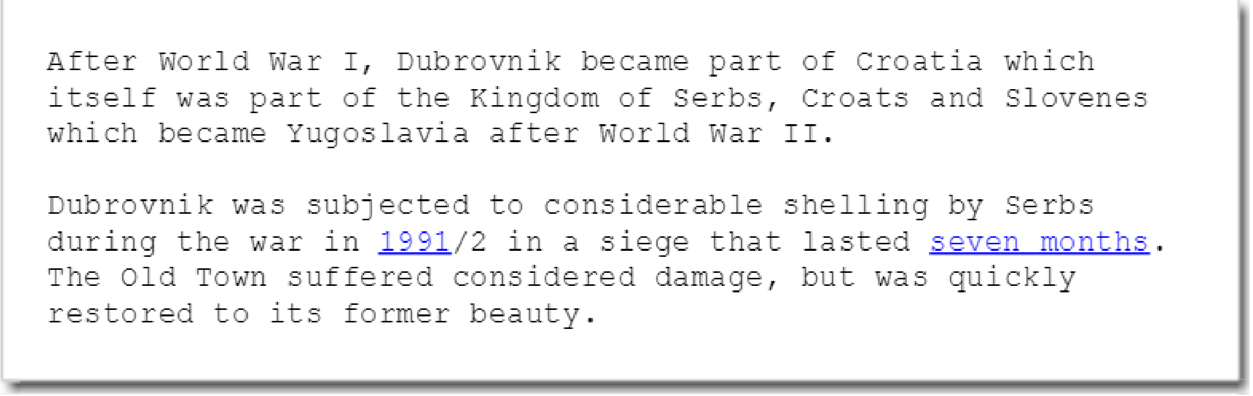
## Dubrovnik in Yugoslavia and Croatia

After World War I, Dubrovnik became part of Croatia which itself was part of the Kingdom of Serbs, Croats and Slovenes which became Yugoslavia after World War II.

Dubrovnik was subjected to considerable shelling by Serbs during the war in 1991/2 in a siege that lasted seven months. The Old Town suffered considered damage, but was quickly restored to its former beauty.

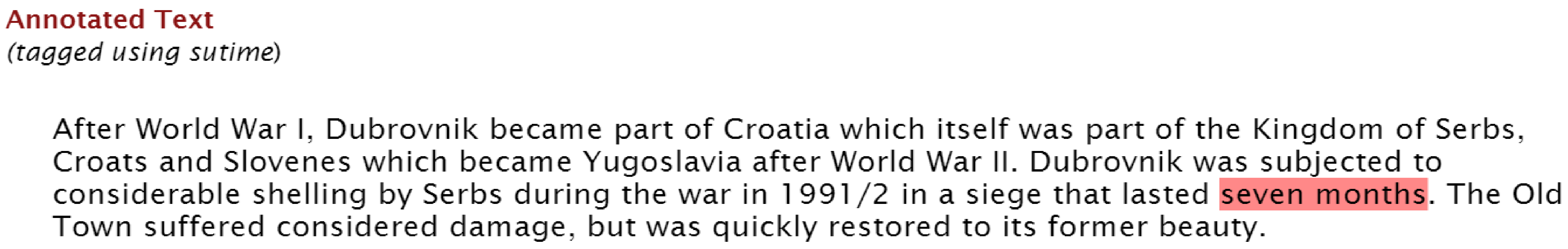
Resulting document:

Tagged using Heideltime



After World War I, Dubrovnik became part of Croatia which itself was part of the Kingdom of Serbs, Croats and Slovenes which became Yugoslavia after World War II.

Dubrovnik was subjected to considerable shelling by Serbs during the war in 1991/2 in a siege that lasted seven months. The Old Town suffered considered damage, but was quickly restored to its former beauty.



**Annotated Text**

*(tagged using sutime)*

After World War I, Dubrovnik became part of Croatia which itself was part of the Kingdom of Serbs, Croats and Slovenes which became Yugoslavia after World War II. Dubrovnik was subjected to considerable shelling by Serbs during the war in 1991/2 in a siege that lasted seven months. The Old Town suffered considered damage, but was quickly restored to its former beauty.

**Any issues? [Compute precision and recall of the extraction results.]**

Source: http://www.visit-croatia.co.uk/index.php/croatia-destinations/dubrovnik/history-dubrovnik/

# Temporal Extraction: Tools

**HeidelTime** (Strötgen 2015)

- Online demo: http://heideltime.ifi.uni-heidelberg.de/heideltime/

**SUTime** (Angel 2012)

- Online demo: http://nlp.stanford.edu:8080/sutime/process

# Relation Extraction

- **Relations** between two or more entities, which relate to one another in real life.
  - A relation is defined in the form of a tuple t = $(e_1, e_2, ..., e_n)$, where $e_i$ are entities in a predefined relation R within document D.
- **Relation extraction**:
  - is a task of detecting relations between entities and assigning relation types to them.
- **Binary relations**: a relation between two entities.
  - located-in(Dubrovnik, Croatia), married-to(Angelina Jolie, Brad Pitt).
- **Higher-order relations**:
  - A 4-ary biomedical point mutation relation: a type of variation, its location, and the corresponding state change from an initial-state to an altered-state.
  - “At codons 12, the occurrence of point mutations from G to T were observed”
  - point mutation(codon, 12, G, T).

# Relation Extraction: Features

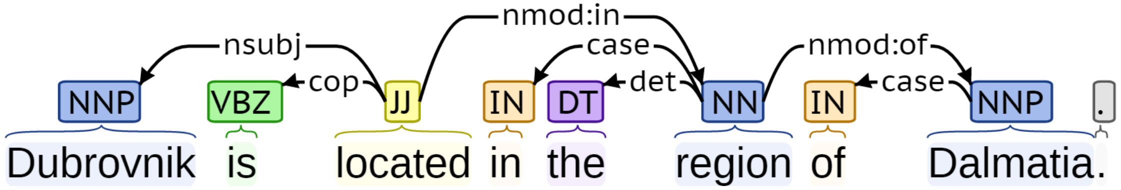
## Syntactic features

| | |
|---|---|
| the entities | [Dubrovnik, the region of Dalmatia] |
| the types of the entities | [Location, Location] |
| word sequence between the entities | [is located in the region of] |
| path length (e.g. the number of words between the entities) | [3] |

## Semantic features

- the path between the two entities in the dependency parse tree

located-in(Dubrovnik, the region of Dalmatia)



# Relation Extraction: Methods

Supervised learning

- E.g. as binary classification

Unsupervised

Semi-supervised and bootstrapping approaches

# Relation Extraction: Supervised Methods

- Relation extraction as a binary classification problem
  - Given a set of features extracted from the sentence *S*, decide if entities in *S* are connected using given relation *R*.
- Disadvantages of supervised methods
  - Need for labelled data. Difficult to extend to new relation types.
  - Extensions to higher order entity relations are difficult as well.
  - Errors in the pre-processing (feature extraction, e.g. parse tree) affect the performance.
  - Pre-defined relations only.

# Relation Extraction: Semi-supervised Methods

- Semi-supervised and bootstrapping approaches (e.g. KnowItAll (Etzioni et al., 2005) and TextRunner (Banko et al., 2007) )
  - “Weak supervision”: Require a small set of tagged seed instances or a few hand-crafted extraction patterns per relation to launch the training process.
  - Use the output of the weak learners as training data for the next iteration.

- **Step1**: Use the seed examples to label some data.
- **Step2**: Induce patterns from the labelled examples.
- **Step3**: Apply the patterns to data, to get a new set of pairs.
- **Return to Step2**, and **iterate** until convergence criteria is reached.

# Relation Extraction: Semi-Supervised Methods

- Relation to be extracted **(author, book)**
- **Step1**: Use the seed examples to label data.
  - Start with one seed (Arthur Conan Doyle, The Adventures of Sherlock Holmes).
  - Pattern [order, author, book, prefix , suffix , middle].
  - Order = 1 if the author string occurs before the book string and 0 otherwise
  - Prefix and suffix are strings of 10 characters to the left/right of the match
  - Middle is the string occurring between the author and book.

Examples from DIPRE (Brin, 1998)

# Relation Extraction: Semi-Supervised Methods

- **Step1 (continued)**: Use the seed examples to label data.
- [order, author, book, prefix , suffix , middle].
- (Arthur Conan Doyle, The Adventures of Sherlock Holmes)
- *S1*=“know that Sir Arthur Conan Doyle *wrote* The Adventures of Sherlock Holmes, in 1892”

# Relation Extraction: Semi-Supervised Methods

- **Step1 (continued)**: Use the seed examples to label data.
- [order, author, book, prefix , suffix , middle].
- (Arthur Conan Doyle, The Adventures of Sherlock Holmes)
- *S1*="know that Sir Arthur Conan Doyle *wrote* The Adventures of Sherlock Holmes, in 1892"
- [1, Arthur Conan Doyle, The Adventures of Sherlock Holmes, know that Sir, in 1892, *wrote*]

# Relation Extraction: Semi-Supervised Methods

- **Step1 (continued)**: Use the seed examples to label data.
- [order, author, book, prefix , suffix , middle].
- (Arthur Conan Doyle, The Adventures of Sherlock Holmes)
- *S1*=“know that Sir Arthur Conan Doyle *wrote* The Adventures of Sherlock Holmes, in 1892”
- [1, Arthur Conan Doyle, The Adventures of Sherlock Holmes, know that Sir, in 1892, *wrote*]
- *S2*=“When Sir Arthur Conan Doyle *wrote* the adventures of Sherlock Holmes in 1892 he was high ...”
- [1, Arthur Conan Doyle, The Adventures of Sherlock Holmes, When Sir, in 1892 he, *wrote*]

# Relation Extraction: Semi-supervised Methods

- **Output Step1:** [order, author, book, prefix , suffix , middle].

  [1, Arthur Conan Doyle, The Adventures of Sherlock Holmes, now that Sir, in 1892, wrote]

  [1, Arthur Conan Doyle, The Adventures of Sherlock Holmes, When Sir, in 1892 he, wrote]
- **Step2:** Induce patterns from the labelled examples.
  - Exact match: generalize the pattern: [Sir, .*?, *wrote*, .*?, in 1892].
  - Approximate match: use similarity metrics for patterns. (Agichtein & Gravano, 2000)
- **Step3:** Apply the patterns to data, to get a new set of pairs.
  - (Arthur Conan Doyle, The Speckled Band).
- **Return to Step2**, and **iterate** until convergence criteria is reached.

# Open Information Extraction (Open IE)

Open IE extracts tuples consisting of argument phrases and a relation between the arguments

- ($arg_1$; rel; $arg_2$).

- For example: *S3*="Trump ($arg_1$) was elected (pred) President ($arg_2$)."
  - (Trump; was elected; President)

Different to relation extraction

- No pre-specified sets of relations
- No domain-specific knowledge engineering

Example applications

- A news reader who wishes to keep abreast of important events
- An analyst who recently acquired a terrorist's laptop

# Open IE: TextRunner

(Banko et al., 2007)

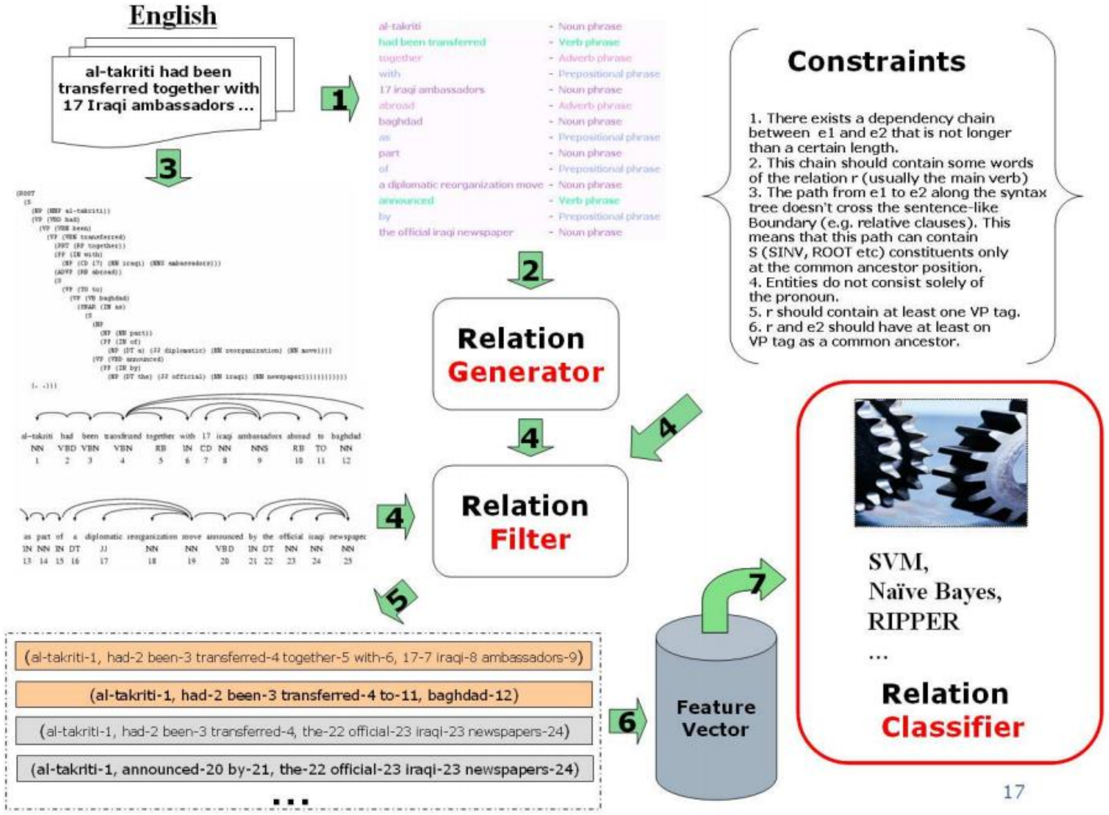
**Relation is a tuple:** $t = (e_1, r, e_2)$

- $e_1$ and $e_2$ are surface forms of entities or noun phrases.
- $r$ denotes a relationship between $e_1$ and $e_2$.

**TextRunner components:**

- Self-supervised Learner: automatic labelling of training data.
- Single-pass Extractor: generates candidate relations from each sentence, runs a classifier and retains the ones labelled as trustworthy relations.
- Redundancy-based Assessor: assigns a probability to each retained tuple based on a probabilistic model of redundancy in text.

# Open IE: TextRunner


English
al-takriti had been transferred together with 17 Iraqi ambassadors …
1
al-takriti - Noun phrase
had been transferred - Verb phrase
together - Adverb phrase
with - Prepositional phrase
17 iraqi ambassadors - Noun phrase
abroad - Adverb phrase
baghdad - Noun phrase
as - Prepositional phrase
part - Noun phrase
of - Prepositional phrase
a diplomatic reorganization move - Noun phrase
announced - Verb phrase
by - Prepositional phrase
the official iraqi newspaper - Noun phrase
2
3
Relation Generator
4
4
4
Relation Filter
5
(al-takriti-1, had-2 been-3 transferred-4 together-5 with-6, 17-7 iraqi-8 ambassadors-9)
(al-takriti-1, had-2 been-3 transferred-4 to-11, baghdad-12)
(al-takriti-1, had-2 been-3 transferred-4, the-22 official-23 iraqi-23 newspapers-24)
(al-takriti-1, announced-20 by-21, the-22 official-23 iraqi-23 newspapers-24)
…
6
Feature Vector
7
SVM,
Naïve Bayes,
RIPPER
…
Relation Classifier
17

**Constraints**

1. There exists a dependency chain between e1 and e2 that is not longer than a certain length.
2. This chain should contain some words of the relation r (usually the main verb)
3. The path from e1 to e2 along the syntax tree doesn't cross the sentence-like Boundary (e.g. relative clauses). This means that this path can contain S (SINV, ROOT etc) constituents only at the common ancestor position.
4. Entities do not consist solely of the pronoun.
5. r should contain at least one VP tag.
6. r and e2 should have at least on VP tag as a common ancestor.

Figure 2: Self-supervised training of Learner module in TextRunner.

(Banko et al., 2007) image: (Bach 2013)

**7-Step training** for each sentence:

**Step 1**: a noun phrase chunker.

**Step 2**: the relation candidator.

**Steps 3-5**: a syntactic parser and dependency parser are run. The relation filter uses parse trees, dependency trees, and set of constraints to label trustworthy relations.

**Step 6**: map each relation to a feature vector representation.

**Step 7**: train a binary classifier using labelled trustworthy and untrustworthy relations.

# Open IE: REVERB and Ollie

REVERB (Fader 2011) and Ollie (Mausam 2012) extract binary relationships from English sentences.

Designed for Web-scale information extraction, where the target relations cannot be specified in advance and speed is important.

REVERB extracts relations mediated by verbs, does not consider the context

- shallow syntactic processing to identify relation phrases that begin with a verb and occur between the argument phrases.

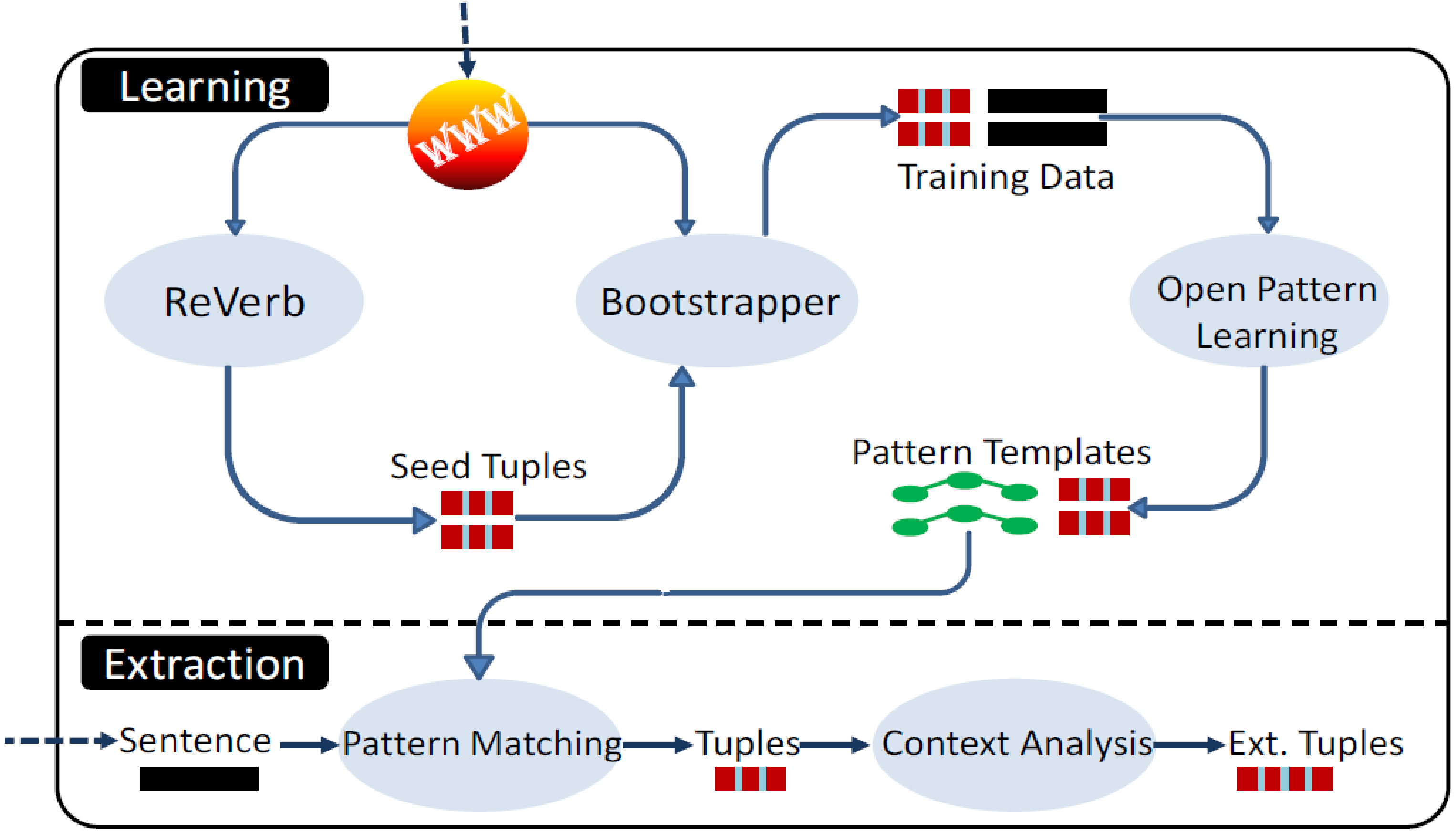
Ollie extracts relations mediated by nouns, adjectives, and more.

Ollie includes contextual information from the sentence in the extractions.

http://reverb.cs.washington.edu/

https://knowitall.github.io/ollie/

# Open IE: Ollie System Architecture



- Use a set tuples from REVERB to bootstrap a large training set.
- Learn open pattern templates over this training set.
- Apply pattern templates at extraction time.
- Analyse the context around the tuple to add information (attribution, clausal modifiers) and a confidence function.

Image (Mausam 2012)

# Open IE: Ollie Bootstrapping

The goal is to automatically create a large training set, which encapsulates the multitudes of ways in which information is expressed in text.

Almost every relation can also be expressed via a REVERB-style verb-based expression.

Retrieve all sentences in a Web corpus that contain all content words in the tuple.

Assumption: sentences express the relation of the original seed tuple.

Not always true:

- (Boyle; is born in; Ireland)
- “Felix G. Wharton was born in Donegal, in the northwest of Ireland, a county where the Boyles did their schooling.”

# Open IE: Ollie Bootstrapping

Over 110,000 seed tuples – high confidence REVERB extractions from ClueWeb - a large Web corpus. Contain only proper nouns in the arguments.

- Seed: “Paul Annacone is the coach of Federer.” ->
- REVERB pattern: (Paul Annacone; is the coach of; Federer).
- Retrieved sentence: “Now coached by Annacone, Federer is winning more titles than ever.”

Enforce additional dependency restrictions on the sentences to reduce bootstrapping errors.

Restrict linear path length between argument and relation in the dependency parse (max 4).

# REVERB vs. Ollie

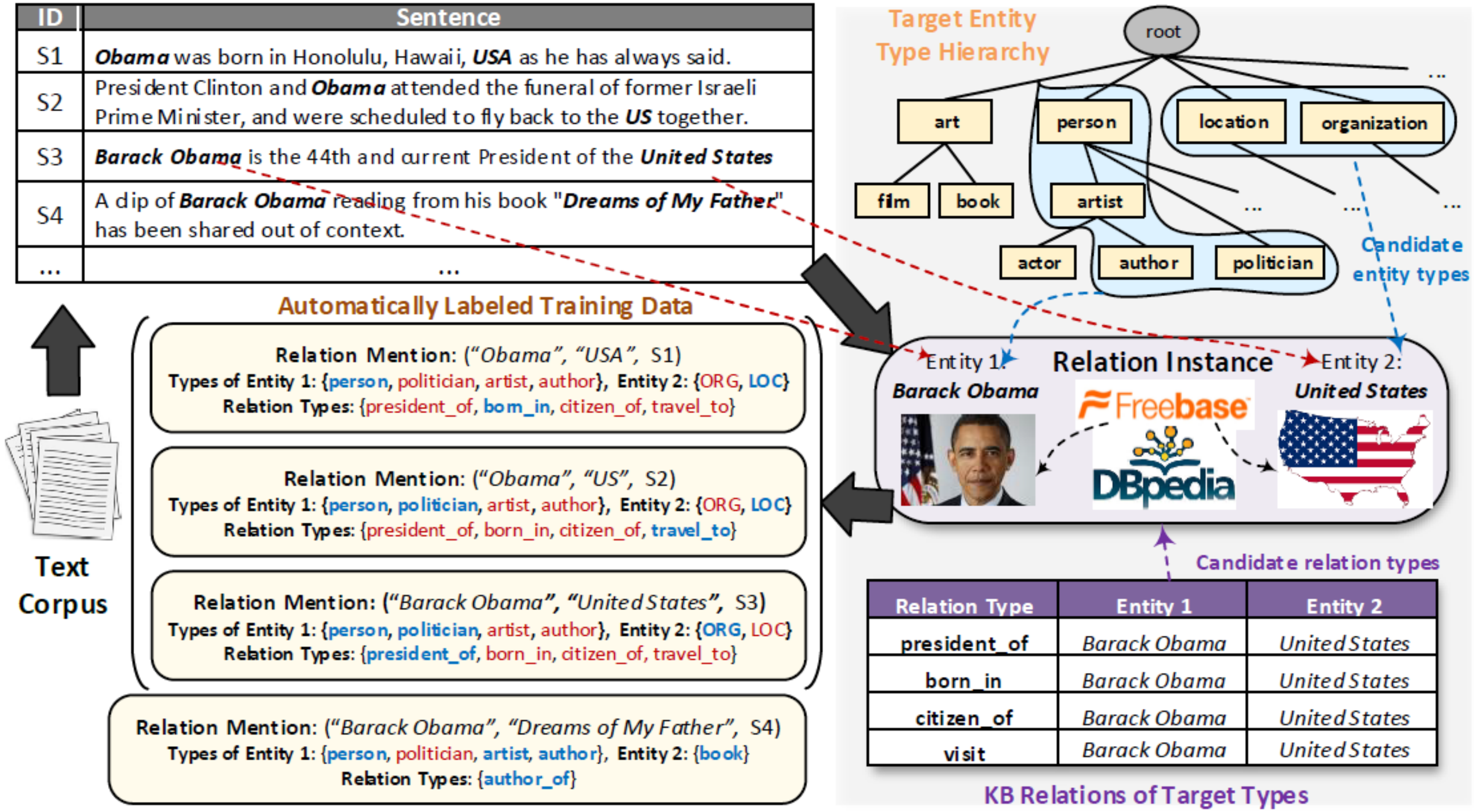
"Early astronomers believed that the earth is the center of the universe."

- R: (the earth; be the center of; the universe)
- O: ((the earth; be the center of; the universe) AttributedTo believe; Early astronomers)

"If he wins five key states, Romney will be elected President."

- R: (Romney; will be elected; President)
- O: ((Romney; will be elected; President) ClausalModifier if; he wins five key states)

# Relation Extraction with Distant Supervision



| ID | Sentence |
|---|---|
| S1 | ***Obama*** was born in Honolulu, Hawaii, ***USA*** as he has always said. |
| S2 | President Clinton and ***Obama*** attended the funeral of former Israeli Prime Minister, and were scheduled to fly back to the ***US*** together. |
| S3 | ***Barack Obama*** is the 44th and current President of the ***United States*** |
| S4 | A clip of ***Barack Obama*** reading from his book "***Dreams of My Father***" has been shared out of context. |
| ... | ... |

**Text Corpus**

**Automatically Labeled Training Data**

**Relation Mention:** (*"Obama"*, *"USA"*, S1)
**Types of Entity 1:** {person, politician, artist, author}, **Entity 2:** {ORG, LOC}
**Relation Types:** {president_of, born_in, citizen_of, travel_to}

**Relation Mention:** (*"Obama"*, *"US"*, S2)
**Types of Entity 1:** {person, politician, artist, author}, **Entity 2:** {ORG, LOC}
**Relation Types:** {president_of, born_in, citizen_of, travel_to}

**Relation Mention:** (*"Barack Obama"*, *"United States"*, S3)
**Types of Entity 1:** {person, politician, artist, author}, **Entity 2:** {ORG, LOC}
**Relation Types:** {president_of, born_in, citizen_of, travel_to}

**Relation Mention:** (*"Barack Obama"*, *"Dreams of My Father"*, S4)
**Types of Entity 1:** {person, politician, artist, author}, **Entity 2:** {book}
**Relation Types:** {author_of}

| Relation Type | Entity 1 | Entity 2 |
|---|---|---|
| president_of | *Barack Obama* | *United States* |
| born_in | *Barack Obama* | *United States* |
| citizen_of | *Barack Obama* | *United States* |
| visit | *Barack Obama* | *United States* |

KB Relations of Target Types

Image: (Xiang Ren, et al., WWW 2017)

The typical workflow:

(1) Detect entity mentions
(2) Perform entity linking
(3) Include all KB types of the KB-mapped entity;
(4) Include all KB relation types between the KB-mapped entities.

Use the automatically labeled training corpus to infer types of the unlinkable candidate mentions.

Problems:
- Domain restriction
- Error propagation
- Label noise

# Joint Extraction of typed Entities and Relations with Distant Supervision

Goal: jointly extract entities and relations of target types with minimal or no human supervision.

Approach:

1) Detect candidate entity mentions with distant supervision and POS tagging
2) Model the mutual constraints between the types of the relation mentions and the types of their entity arguments

(3) Rank relevant relation types (as opposed to every candidate type considered relevant to the mention).

(Xiang Ren, et al., WWW 2017)

# NLP & ML Software

**Natural Language Processing**:

- Stanford NLP (Java)
- GATE (general purpose architecture, includes other NLP and ML software as plugins)
- OpenNLP (Java)
- NLTK (Python)

**Machine Learning**:

- scikit-learn (Python)
- Mallet (Java)
- WEKA (Java)
- Alchemy (graphical models, Java)
- FACTORIE, wolfe (graphical models, Scala)
- CRFSuite (efficient implementation of CRFs, Python)
- Apache Spark Mllib
- Apache Mahout

# NLP & ML Software

**Ready to use NERC software**:

- ANNIE (rule-based, part of GATE)
- Wikifier (based on Wikipedia)
- FIGER (based on Wikipedia, fine-grained Freebase NE classes)

**Almost ready to use NERC software**:

- CRFSuite (already includes Python implementation for feature extraction, you just need to feed it with training data, which you can also download)

**Ready to use RE software**:

- ReVerb, Ollie (Open IE, extract patterns for any kind of relation)
- MultiR (Distant supervision, relation extractor trained on Freebase)

**Web content extraction software**:

- Boilerpipe (extract main text content from Web pages)
- Jsoup (traverse elements of Web pages individually, also allows to extract text)
- iCrawl (focused web crawling) http://icrawl.l3s.uni-hannover.de/

# Summary

- In this session we provided an overview of the state-of-the-art Information Extraction methods for:
    - Named Entity Extraction
    - Named Entity Linking
    - Temporal Extraction
    - Relation Extraction

- We addressed
    - Rule-based approaches
    - Machine learning approaches
    - Different supervision models for machine learning
    - An overview of tools

- We will get familiar with selected extraction & linking tools in the hands-on session

# Thank you!

# Questions, Comments?

Dr. Elena Demidova

L3S Research Center
Leibniz University of Hannover

email: demidova@L3S.de
www: https://demidova.wordpress.com

This presentation contains photos from www.fotolia.de

# References

**Entity Extraction and Linking**

- Roman Prokofyev, Gianluca Demartini, Philippe Cudré-Mauroux (2014): Effective named entity recognition for idiosyncratic web collections. WWW 2014: 397-408.
- H. Dai, C. Wu, R. Tsai, and W. Hsu. From Entity Recognition to Entity Linking: A Survey of Advanced Entity Linking Techniques. In The 26th Annual Conference of the Japanese Society for Artificial Intelligence, pp 1–10, 2012.
- Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. 2005. Unsupervised named-entity extraction from the web: an experimental study. Artif. Intell. 165, 1 (June 2005), 91-134.

**Temporal extraction**

- Pustejovsky, J., Knippen, R., Littman, J., & Sauri, R. (2005). Temporal and event information in natural language text. Language resources and evaluation, 39(2–3), 23–164. 2005.
- Jannik Strötgen, Michael Gertz. Multilingual and cross-domain temporal tagging. Language Resources and Evaluation. June 2013, Volume 47, Issue 2, pp 269-298.
- Jannik Strötgen, Michael Gertz. A Baseline Temporal Tagger for All Languages. EMNLP'15.
- Angel X. Chang and Christopher D. Manning. 2012. SUTIME: A Library for Recognizing and Normalizing Time Expressions. 8th International Conference on Language Resources and Evaluation (LREC 2012).

# References

- **Relation extraction and Open IE**

  - N Bach, S Badaskar. A review of relation extraction. 2013
  - Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007). Open information extraction from the web. Proceedings of IJCAI ’07.
  - Agichtein, E., & Gravano, L. (2000). Snowball: Extracting relations from large plain-text collections. Proceedings of the Fifth ACM International Conference on Digital Libraries.
  - Brin, S. (1998). Extracting patterns and relations from the world wide web. WebDB Workshop at 6th International Conference on Extending Database Technology, EDBT ’98.
  - Mausam, M. S., et al. 2012. Open language learning for information extraction. EMNLP-CoNLL '12.
  - Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In EMNLP '11. pp. 1535-1545.
  - Xiang Ren, et al. 2017. CoType: Joint Extraction of Typed Entities and Relations with Knowledge Bases. In *Proc. of the* WWW '17.

# References

- **Relation extraction**

  - Xiang Ren, et al. 2017. CoType: Joint Extraction of Typed Entities and Relations with Knowledge Bases. In *Proc. of the* WWW '17.
  - Nicolas Heist and Heiko Paulheim. Language-agnostic Relation Extraction from Wikipedia Abstracts. In Proc. of the ISWC 2017.
  - T. Mitchell, et al. Never-Ending Learning. In Proceedings of the Conference on Artificial Intelligence (AAAI), 2015