Named Entity Extraction
-
Overview

PD Dr. Günter Neumann
DFKI and Saarland University

Named Entity Recognition (NER)

- NER aims at finding unknown named entities in unstructured text.
- A promising approach is to automatically expand sets of known (usually few) entities.

Named Entity Identification (NEI)

- NEI aims at finding known named entities in unstructured text.

- A promising approach is finding approximate matches in a text with respect to a large dictionary of known entities

  $\Longrightarrow$ ***approximate dictionary matching***

Named Entity Disambiguation (NED)

- NED aims at disambiguating the known referents of a named entity in unstructured text.

- A promising approach is to identify the similarity of textual context of a named entity with attribute descriptions of candidate referents.

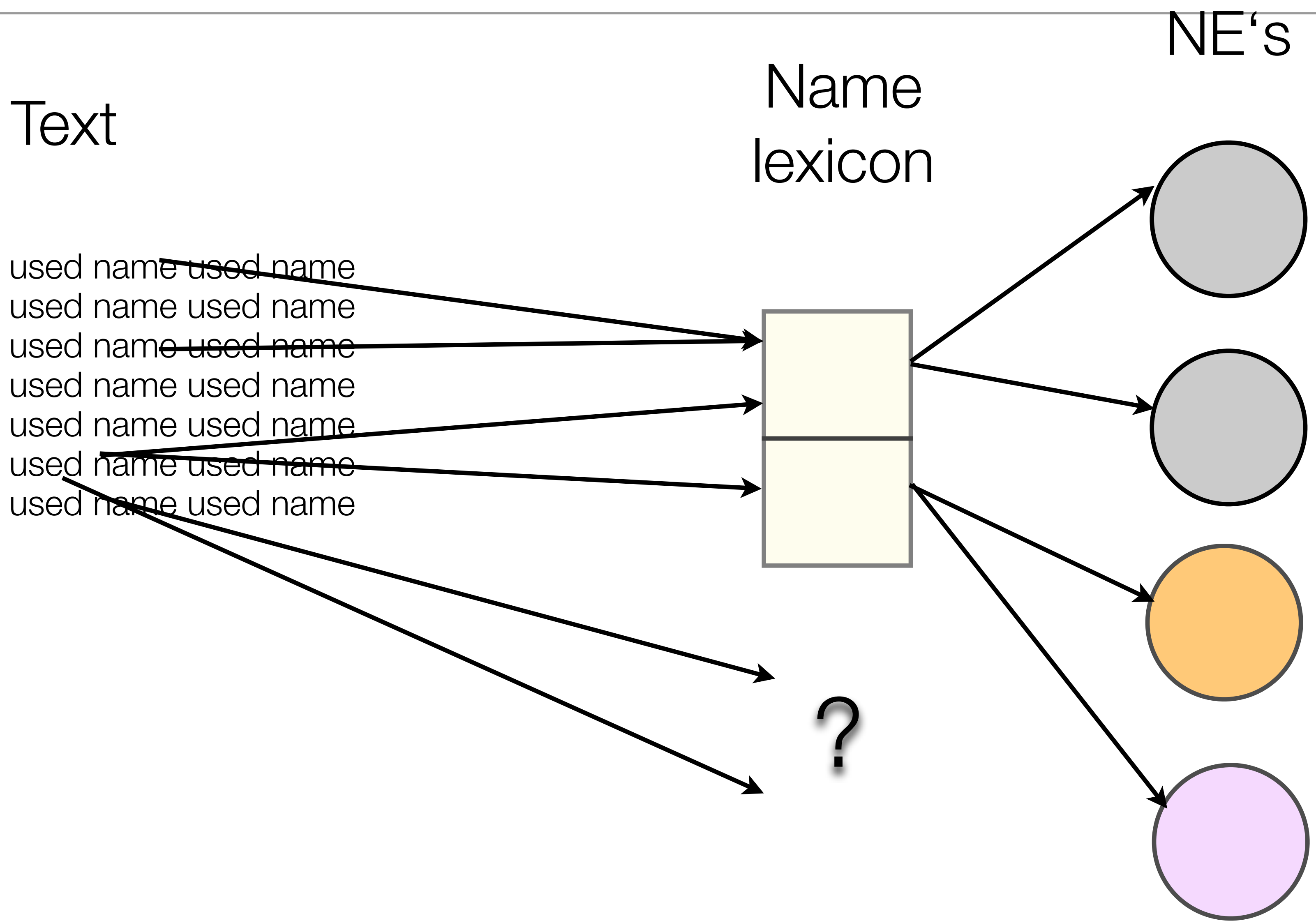
What is a NE ?

- A NE is a subclass of the entities, for which one or many rigid descriptors stands for the referent.
- Rigid descriptors are proper nouns and certain natural kind terms.
- NE entity processing involves the
  - recognition that a text string is a rigid descriptor
  - the determination of its (major) type
  - the disambiguation of the NE term

Semantic Perspective to NE‘s

- An entity is a unique object that exists as a particular and distinct unit (living or non-living).
- Semantically an entity can be represented as a set of attribute value pairs.
- Named entity is a linguistic object for denoting the name and type of an entity.
- Mentions are definite descriptions for denoting (a bundle) of attribute value pairs of an entity.

Why is it difficult ?



The who, where, when & how much in a sentence

- The task: identify lexical and phrasal information in text which express references to named entities NE, e.g.,
  - person names
  - company/organization names
  - locations
  - dates&times
  - percentages
  - monetary amounts
- Determination of an NE
  - Specific type according to some taxonomy
  - Canonical representation (template structure)

Example of NE-annotated text

Delimit the named entities in a text and tag them with NE types:

<ENAMEX TYPE=„LOCATION“>Italy</ENAMEX>‘s business world was rocked by the announcement <TIMEX TYPE=„DATE“>last Thursday</TIMEX> that Mr. <ENAMEX TYPE=„PERSON“>Verdi</ENAMEX> would leave his job as vice-president of <ENAMEX TYPE=„ORGANIZATION“>Music Masters of Milan, Inc</ENAMEX> to become operations director of <ENAMEX TYPE=„ORGANIZATION“>Arthur Andersen</ENAMEX>.

„Milan“ is part of organization name

„Arthur Andersen“ is a company

„Italy“ is sentence-initial ⇒ capitalization useless

NE and Question-Answering

- Often, the expected answer type of a question is a NE

  What was the name of the first Russian astronaut to do a spacewalk?

  - Expected answer type is PERSON

  Name the five most important software companies!

  - Expected answer type is a list of COMPANY

  Where is does the ESSLLI 2004 take place?

  - Expected answer type is LOCATION (subtype COUNTRY or TOWN)

  When will be the next talk?

  - Expected answer type is DATE

NE Co-reference

Norman Augustine ist im Grunde seines Herzens ein friedlicher Mensch."Ich könnte niemals auf irgend etwas schiessen", versichert der 57jährige Chef des US-Rüstungskonzerns Martin Marietta Corp. (MM). ... Die Idee zu diesem Milliardendeal stammt eigentlich von GE-Chef JohnF. Welch jr. Er schlug Augustine bei einem Treffen am 8. Oktober den Zusammenschluss beider Unternehmen vor. Aber Augustine zeigte wenig Interesse, Martin Marietta von einem zehnfach grösseren Partner schlucken zu lassen.

- Martin Marietta can be a person name or a reference to a company
- If MM is not part of an abbreviation lexicon, how do we recognize it?
  - Also by taking into account NE reference resolution.

NE is an interesting problem

- Productivity of name creation requires lexicon free pattern recognition
- NE ambiguity requires resolution methods
- Fine-grained NE classification needs fined-grained decision making methods
  - Taxonomy learning
- Multi-linguality
  - A text might contain NE expressions from different languages

Basic Problems in NE

- Variation of NEs – e.g. John Smith, Mr Smith, John.
- Ambiguity of NE types: John Smith (company vs. person)
  – May (person vs. month)
  – Washington (person vs. location)
  – 1945 (date vs. time)
- Ambiguity with common words, e.g. "may"

More complex problems in NE

- Issues of style, structure, domain, genre etc.
- Punctuation, spelling, spacing, formatting, ... all have an impact:

```
Dept. of Computing and Maths

Manchester Metropolitan University

Manchester

United Kingdom
```

```
➢ Tell me more about Leonardo

➢ Da Vinci
```

Two principle ways of specifying NE

- Hand-craft rule writing
  - still the best performance when fined-grained classification is needed
  - Hard to adapt to new domains
- Machine learning
  - System-based adaptation two new domains
  - Very good for coarse-grained classification
  - Still require large training data

List lookup approach - baseline

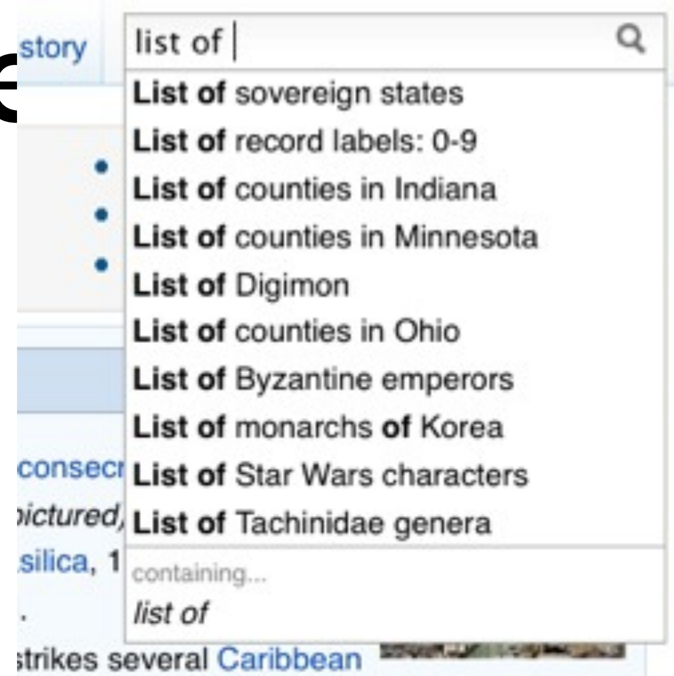
- System that recognizes only entities stored in its lists (gazetteers).
- Advantages - Simple, fast, language independent, easy to retarget (just create lists)
- Disadvantages - collection and maintenance of lists, cannot deal with name variants, cannot resolve ambiguity
- But see: approximate dictionary lookup !

From Cunningham & Bontcheva, 2003

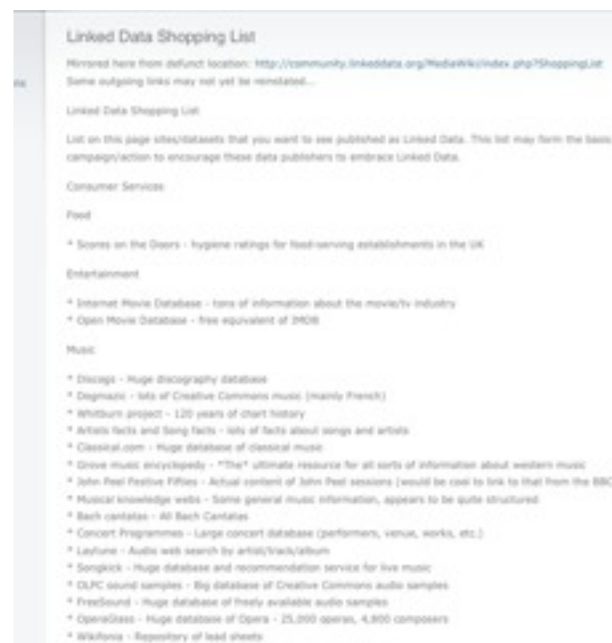
Creating Gazetteer Lists

- structured data sources
  - Online phone directories and yellow pages for person and organisation names
  - U.S. census bureau
    - http://www.census.gov/genealogy/www/data/1990surnames/
  - Locations lists
    - US GEOnet Names Server (GNS) data – 3.9 million locations with 5.37 million names
      - http://earth-info.nga.mil/gns/html/
    - The World Gazetteer provides a comprehensive set of population data and related statistics
      - http://www.world-gazetteer.com/

Creating Gazette



- semi-structured data sources
  - Wikipedia
  - Linked data
    - http://linkeddata.org/home
- To extract gazetteers from these sources wrapper technology is needed
- Automatic methods for extracting gazetteers via Machine Learning



The hand-crafted approach

Uses hand-written context-sensitive reduction rules:

1) title capitalized word => title person_name
   compare „Mr. Jones“ vs. „Mr. Ten-Percent“
   => no rule without exceptions

2) person_name, „the“ adj* „CEO of“ organization
   „Fred Smith, the young dynamic CEO of BlubbCo“
   => ability to grasp non-local patterns

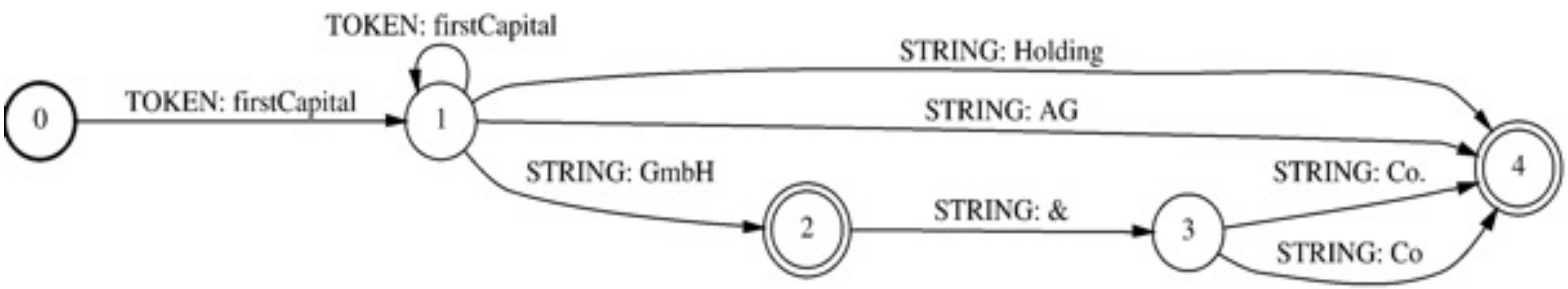
3) plus help from databases of known named entities

Named Entity Finder SPPC (cf. Neumann & Piskorski, 2002)

Arcs of the WFSAs are predicates on lexical items:

(a) **STRING: s**, holds if the surface string mapped by current lexical item is of the form **s**
(b) **STEM: s**, holds if: the current lexical item has a prefered reading with stem **s** or the current lexical item does not have prefered reading, but at least one reading with stem **s**
(c) **TOKEN: x**, holds if the token type of the surface string mapped by current lexical item is **x**

Example: simple automaton for recognition of company names



additional constraint: disallow determiner reading for the first word
candidate: „Die Braun GmbH & Co.“ extracted: „Braun GmbH & Co.“

Evaluation of SPPC*

	Number of NEs				
NE-Type	correct	wrong	missing	Precision	Recall
organisation	745	53	196	93%	80%
person	180	16	22	92%	90%
location	497	10	81	98%	86%
all	1422	79	299	95%	83%
nouns	1456	78	217	95%	88%

Manual check with 100 annotated test documents

Good performance for the recognition of NEs and generic nouns (including compound analysis)

problems with English NEs ► upgrade lexicon

* Markus Heidmann, Master Thesis, AIFB&Ontoprise, 2001

Analysis of company names

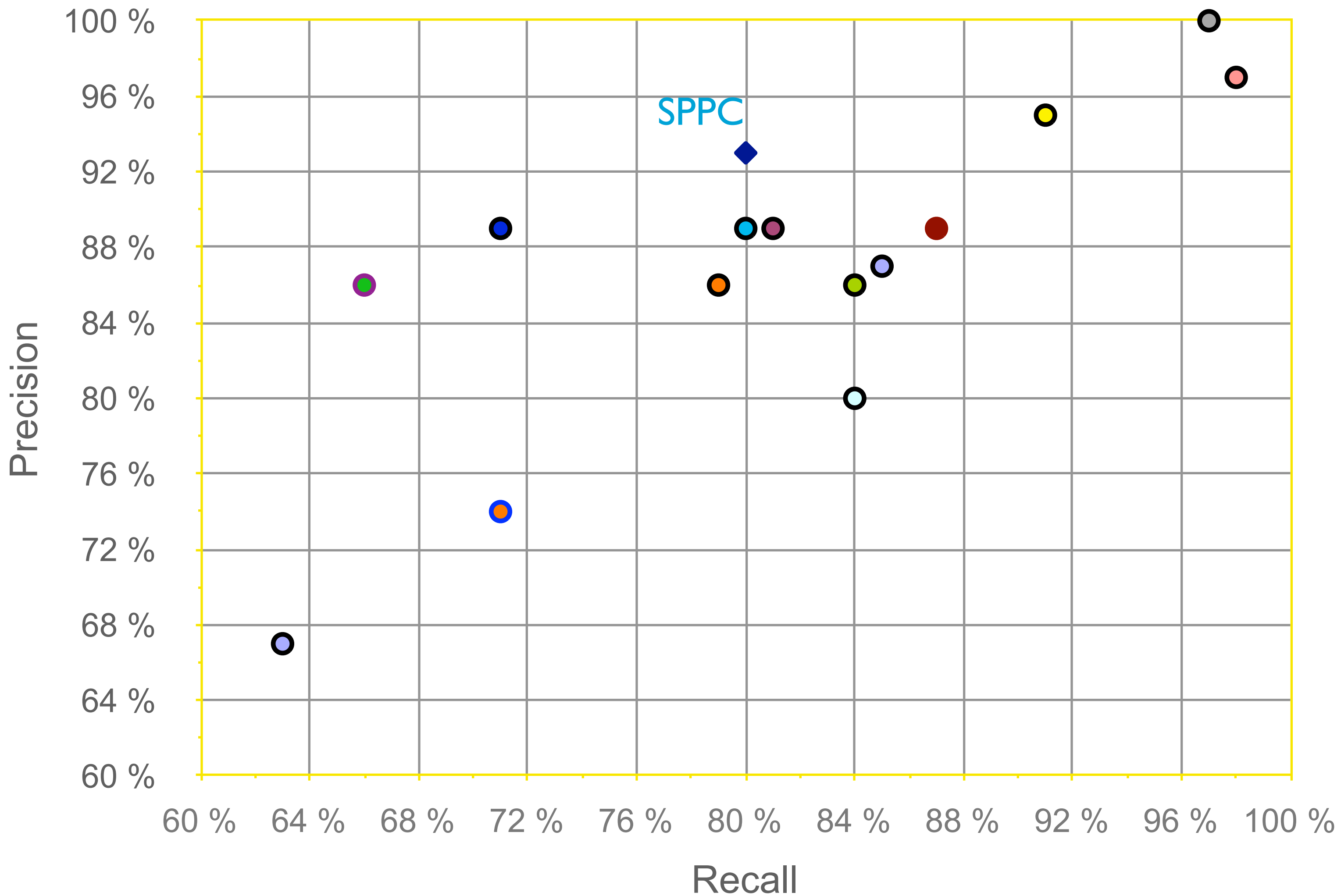
	Number of NEs				
Type	correct	wrong	missing	Precision	Recall
DAX	13	2	15	86%	50%
Dow Jones	8	1	21	88%	30%
Nemax 50	8	15	27	35%	46%
Nemax 50	80	28	2	74%	98%
Euro-Stoxx-50	15	8	27	65%	46%

Problems with the recognition of compound company names if a one part matches with a generic word (e.g., Münchener Rück, MAN)

SPPC company gazetteer too small

high recall for companies through NE reference resolution

SPPC NE company recognition results compared to MUC-7 systems (only indicative!)



Systems of MUC-7 (English)

Problems with the shallow parsing approach

- Ambiguously capitalised words (first word in sentence)
  `[All American Bank]` vs. All `[State Police]`

- Semantic ambiguity
  `"John F. Kennedy"` = airport (location)
  `"Philip Morris"` = organisation

- Structural ambiguity
  `[Cable and Wireless]` vs. `[Microsoft]` and `[Dell]`;
  `[Center for Computational Linguistics]` vs. message from `[City Hospital]` for `[John Smith]`

From Cunningham & Bontcheva, 2003

Shallow Parsing Approach with Context

- Use of context-based patterns is helpful in ambiguous cases
- "David Walton" and "Goldman Sachs" are indistinguishable
- But with the phrase "David Walton of Goldman Sachs" and the Person entity "David Walton" recognised, we can use the pattern "[Person] of [Organization]" to identify "Goldman Sachs“ correctly.

From Cunningham & Bontcheva, 2003

Identification of Contextual Information

- Use KWIC (KeyWord In Context) index and concordancer to find windows of context around entities
- Search for repeated contextual patterns of either strings, other entities, or both
- Manually post-edit list of patterns, and incorporate useful patterns into new rules
- Repeat with new entities

From Cunningham & Bontcheva, 2003

Examples of context patterns

- [PERSON] earns [MONEY]
- [PERSON] joined [ORGANIZATION]
- [PERSON] left [ORGANIZATION]
- [PERSON] joined [ORGANIZATION] as [JOBTITLE]
- [ORGANIZATION]'s [JOBTITLE] [PERSON]
- [ORGANIZATION] [JOBTITLE] [PERSON]
- the [ORGANIZATION] [JOBTITLE]
- part of the [ORGANIZATION]
- [ORGANIZATION] headquarters in [LOCATION]
- price of [ORGANIZATION]
- sale of [ORGANIZATION]
- investors in [ORGANIZATION]
- [ORGANIZATION] is worth [MONEY]
- [JOBTITLE] [PERSON]
- [PERSON], [JOBTITLE]

From Cunningham & Bontcheva, 2003

Why Machine Learning NE?

- System-based adaptation two new domains
  - Fast development cycle
  - Manual specification too expensive
  - Language-independence of learning algorithms
  - NL-tools for feature extraction available, often as open-source
- Current approaches already show near-human-like performance
  - Can easily be integrated with externally available Gazetteers
- High innovation potential
  - Core learning algorithms are language independent, which supports multi-linguality
  - Novel combinations with relational learning approaches
  - Close relationship to currently developed ML-approaches of reference resolution

Machine Learning Approaches

- ML approaches frequently break down the NE task in two parts:
  - Recognising the entity boundaries
  - Classifying the entities in the NE categories
- Some work is only on one task or the other
- Tokens in text are often coded with the IOB scheme
  - O – outside, B-XXX – first word in NE, I-XXX – all other words in NE
  - Easy to convert to/from inline MUC-style markup
  -
Argentina	B-LOC
played	O
with	O
Del	B-PER
Bosque	I-PER

From Cunningham & Bontcheva, 2003

Different Strategies

- Supervised learning
  - Training is based on available very large annotated corpus
  - Mainly statistical-based methods used
    - HMM, MEM, connectionists models, SVM, CRF, hybrid ML-methods (cf. http://cnts.uia.ac.be/conll2003/ner/)

Different Strategies

- Semi-supervised learning
  - Main technique is called „bootstrapping“
  - Training only needs very few seeds and very large un-annotated corpus

Different Strategies

- Unsupervised learning
  - The typical approach in unsupervised learning is clustering
  - gather named entities from clustered groups based on the similarity of context
  - labeling of identified NEs with help of generic semantic lexicons (e.g., word net) or NE-specific Hearst-patterns like „“city such as”, “organization such as”, etc.

Different Feature Sets

- Different degree of NL-preprocessing
  - Character-level features (Whitelaw&Patrick, CoNLL, 2003)
  - Tokenization (Bikel et al., ANLP 1997)
  - POS + lemmatization (Yangarber et al. Coling 2002)
  - Morphology (Cucerzan&Yarowsky, EMNLP 1999)
  - Full parsing (Collins&Singer, EMNLP 1999)

Word-Level features (cf. Nadeau & Sekine, 2007)

Features	Examples
Case	- Starts with a capital letter - Word is all uppercased - The word is mixed case (e.g., ProSys, eBay)
Punctuation	- Ends with period, has internal period (e.g., St., I.B.M.) - Internal apostrophe, hyphen or ampersand (e.g., O'Connor)
Digit	- Digit pattern (*see section 3.1.1*) - Cardinal and Ordinal - Roman number - Word with digits (e.g., W3C, 3M)
Character	- Possessive mark, first person pronoun - Greek letters
Morphology	- Prefix, suffix, singular version, stem - Common ending (*see section 3.1.2*)
Part-of-speech	- proper name, verb, noun, foreign word
Function	- Alpha, non-alpha, n-gram (*see section 3.1.3*) - lowercase, uppercase version - pattern, summarized pattern (*see section 3.1.4*) - token length, phrase length

List look-up features (cf. Nadeau & Sekine, 2007)

Features	Examples
General list	- General dictionary (see section 3.2.1) - Stop words (function words) - Capitalized nouns (e.g., January, Monday) - Common abbreviations
List of entities	- Organization, government, airline, educational - First name, last name, celebrity - Astral body, continent, country, state, city
List of entity cues	- Typical words in organization (see 3.2.2) - Person title, name prefix, post-nominal letters - Location typical word, cardinal point

Document features (cf. Nadeau & Sekine, 2007)

Features	Examples
Multiple occurrences	- Other entities in the context - Uppercased and lowercased occurrences (see 3.3.1) - Anaphora, coreference (see 3.3.2)
Local syntax	- Enumeration, apposition - Position in sentence, in paragraph, and in document
Meta information	- Uri, Email header, XML section, (see section 3.3.3) - Bulleted/numbered lists, tables, figures
Corpus frequency	- Word and phrase frequency - Co-occurrences - Multiword unit permanency (see 3.3.4)

Performance of supervised methods (CoNLL, 2003)

English	precision	recall	F
[FIJZ03]	88.99%	88.54%	88.76±0.7
[CN03]	88.12%	88.51%	88.31±0.7
[KSNM03]	85.93%	86.21%	86.07±0.8
[ZJ03]	86.13%	84.88%	85.50±0.9
[CMP03b]	84.05%	85.96%	85.00±0.8
[CC03]	84.29%	85.50%	84.89±0.9
[MMP03]	84.45%	84.90%	84.67±1.0
[CMP03a]	85.81%	82.84%	84.30±0.9
[ML03]	84.52%	83.55%	84.04±0.9
[BON03]	84.68%	83.18%	83.92±1.0
[MLP03]	80.87%	84.21%	82.50±1.0
[WNC03]*	82.02%	81.39%	81.70±0.9
[WP03]	81.60%	78.05%	79.78±1.0
[HV03]	76.33%	80.17%	78.20±1.0
[DD03]	75.84%	78.13%	76.97±1.2
[Ham03]	69.09%	53.26%	60.15±1.3
baseline	71.91%	50.90%	59.61±1.2

German	precision	recall	F
[FIJZ03]	83.87%	63.71%	72.41±1.3
[KSNM03]	80.38%	65.04%	71.90±1.2
[ZJ03]	82.00%	63.03%	71.27±1.5
[MMP03]	75.97%	64.82%	69.96±1.4
[CMP03b]	75.47%	63.82%	69.15±1.3
[BON03]	74.82%	63.82%	68.88±1.3
[CC03]	75.61%	62.46%	68.41±1.4
[ML03]	75.97%	61.72%	68.11±1.4
[MLP03]	69.37%	66.21%	67.75±1.4
[CMP03a]	77.83%	58.02%	66.48±1.5
[WNC03]	75.20%	59.35%	66.34±1.3
[CN03]	76.83%	57.34%	65.67±1.4
[HV03]	71.15%	56.55%	63.02±1.4
[DD03]	63.93%	51.86%	57.27±1.6
[WP03]	71.05%	44.11%	54.43±1.4
[Ham03]	63.49%	38.25%	47.74±1.5
baseline	31.86%	28.89%	30.30±1.3

http://www.cnts.ua.ac.be/conll2003/ner/

Main features used by CoNLL 2003 systems

	lex	pos	aff	pre	ort	gaz	chu	pat	cas	tri	bag	quo	doc
Florian	+	+	+	+	+	+	+	-	+	-	-	-	-
Chieu	+	+	+	+	+	+	-	-	-	+	-	+	+
Klein	+	+	+	+	-	-	-	-	-	-	-	-	-
Zhang	+	+	+	+	+	+	+	-	-	+	-	-	-
Carreras (a)	+	+	+	+	+	+	+	+	-	+	+	-	-
Curran	+	+	+	+	+	+	-	+	+	-	-	-	-
Mayfield	+	+	+	+	+	-	+	+	-	-	-	+	-
Carreras (b)	+	+	+	+	+	-	-	+	-	-	-	-	-
McCallum	+	-	-	-	+	+	-	+	-	-	-	-	-
Bender	+	+	-	+	+	+	+	-	-	-	-	-	-
Munro	+	+	+	-	-	-	+	-	+	+	+	-	-
Wu	+	+	+	+	+	+	-	-	-	-	-	-	-
Whitelaw	-	-	+	+	-	-	-	-	+	-	-	-	-
Hendrickx	+	+	+	+	+	+	+	-	-	-	-	-	-
De Meulder	+	+	+	-	+	+	+	-	+	-	-	-	-
Hammerton	+	+	-	-	-	+	+	-	-	-	-	-	-

Table 3: Main features used by the the sixteen systems that participated in the CoNLL-2003 shared task sorted by performance on the English test data. Aff: affix information (n-grams); bag: bag of words; cas: global case information; chu: chunk tags; doc: global document information; gaz: gazetteers; lex: lexical features; ort: orthographic information; pat: orthographic patterns (like Aa0); pos: part-of-speech tags; pre: previously predicted NE tags; quo: flag signing that the word is between quotes; tri: trigger words.

Learning Approaches in CoNLL

- Most systems used
  - Maximum entropy modeling (5)
  - Hidden-Markov models (4)
  - Connectionists methods (4)
- Near all systems used external resources, e.g., gazetteers
- Best systems performed hybrid learning approach

Hidden Markov Model (HMM) for NE

- Assumption:
  - There exists an underlying finite state machine (not directly observable, hence hidden) that changes state with each input element (words)
  - The probability of a recognized constituent is conditioned not only on the words seen, but the state that the machine is in at that moment.
  - e.g., having observed „John“ then if current word is „Smith“ then sequence „John Smith“ is quite likely a person name, but if current word is „Deere“ then sequence „John Deere“ is quite likely a company name.
- Construction of an HMM
  - constructing a good hidden state model
  - examining enough training data to accurately estimate the probabilities of the various state transitions given sequences of words

HMM for NE

- Hidden state transition model governs word sequences
- Transitions are probabilistic
- Estimate transition probabilities from an annotated corpus
  - $P(s_j|s_{j-1}, w_j)$
- At runtime, compute maximum likelihood path through network
- Viterbi algorithm

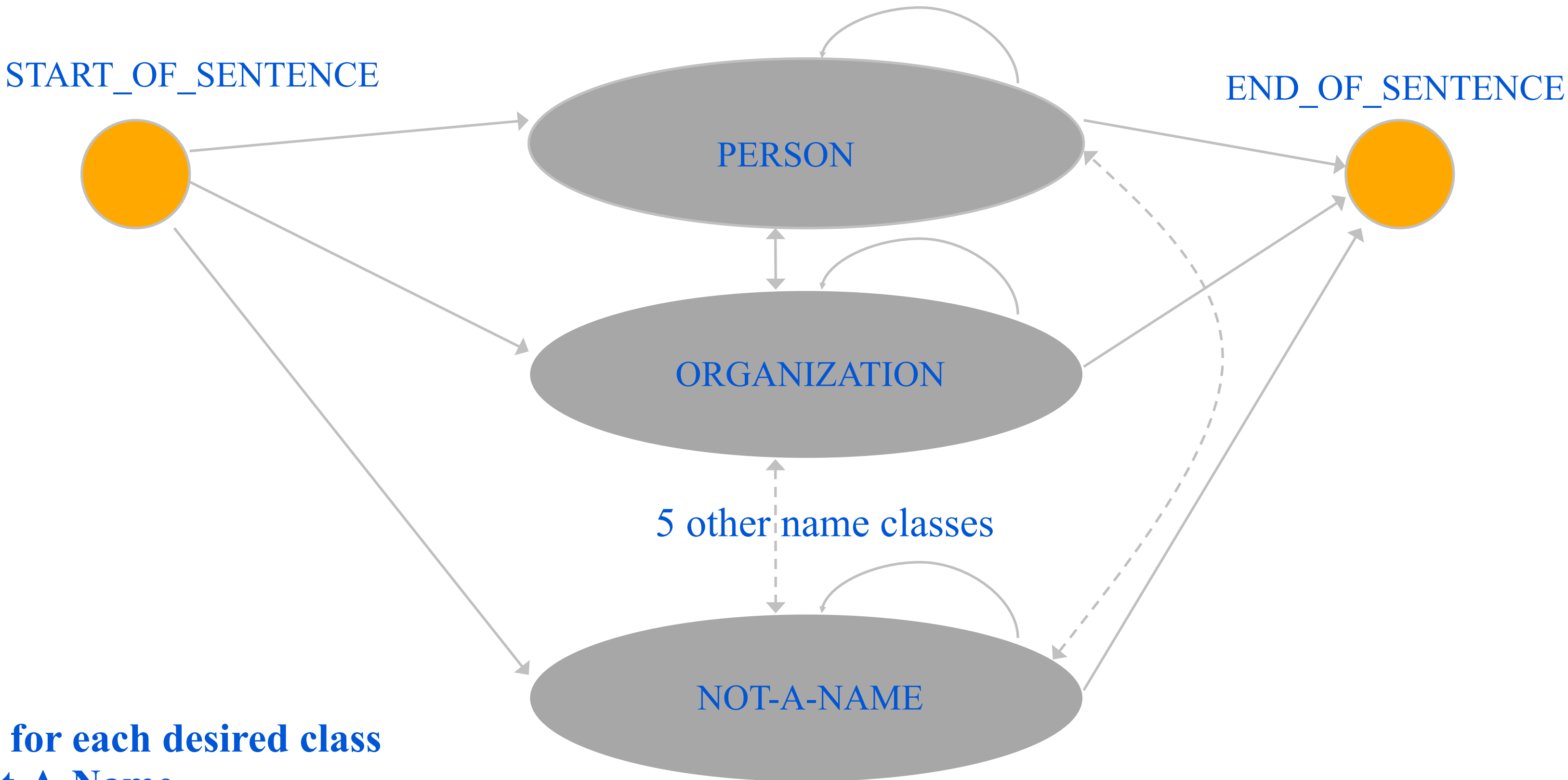
w_{j-1}	w_j
John	Smith
S_{j-1}	S_j
PER	?

IdentiFinder [Bikel et al 99]

- Based on Hidden Markov Models
- Their HMM has 7 regions – one for each MUC type, not-name, begin-sentence and end-sentence
- Features (the only language dependent part)
  - Capitalisation
  - Numeric symbols
  - Punctuation marks
  - Position in the sentence
  - 14 features in total, combining above info, e.g., containsDigitAndDash (09-96), containsDigitAndComma (23,000.00)

From Cunningham & Bontcheva, 2003

HMM for NE



One region for each desired class
One for Not-A-Name
Within each region, a model for computing the likelihood of words occurring within that region
A statistical bigram language model computes the likelihood of a sequence of words by employing a Markov chain, where every word's likelihood is based simply on the previous word.

IdentiFinder (2)

- Back-off models and smoothing
- Unknown words
- Further back-off and smoothing
- Different strategies for name-class bigrams, first-word bigrams and non-first-word bigrams

Example: Handling of unknown words

- Vocabulary is built as it trains
- All unknown words are mapped to the token _UNK_
- _UNK_ can occur
  - As the current word, previous word, or both
- Train an unknown word model on held-out data
  - Gather statistics of unknown words in the midst of known words
- Approach in IdentiFinder
  - 50% hold out for unknown word model
  - Do the same for the other 50%
  - combine bigram counts for the first unknown training file

IdentiFinder - Experiments

- MUC-6 (English) and MET-1(Spanish) corpora used for evaluation
- Mixed case English
  - IdentiFinder - 94.9% f-measure
  - Best rule-based – 96.4%
- Spanish mixed case
  - IdentiFinder – 90%
  - Best rule-based - 93%
  - Lower case names, noisy training data, less training data
- Training data:
  - 650,000 words, but similar performance with half of the data.
  - Less than 100,000 words reduce the performance to below 90% on English

From Cunningham & Bontcheva, 2003