Information Extraction and Question-Answering Systems

Foundations and methods

Dr. Günter Neumann
LT-Lab, DFKI
neumann@dfki.de

What the lecture will cover

- Machine Learning for IE
- Lexical processing
- Evaluation Methods
- Basic Terms & Examples
- Parsing of Unrestricted Text
- Generic NL Core system
- Domain Modelling
- Question/Answering Core components
- Advanced Topics

Contents

- Task & Motivation, example
- Hand-crafted approach
  - TBL (Alembic workbench)
  - DFKI´s SMES technology
- Automated (ML) approaches
  - Hidden Markov Models
  - Decision Trees
  - Maximum Entropy Models
- Hand-crafted vs. automated
- Increasing performance

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's
  - Specific type according to some taxonomy
  - Canonical representation (template structure)

Example from MUC-7

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

Difficulties

- too numerous to include in dictionaries
- changing constantly
- appear in many variant forms
- subsequent occurrences might be abbreviated

⇒ list search/matching doesn‘t perform well

⇒ context based pattern matching needed

Difficulties

Whether a phrase is a proper name, and what name class it has, depends on

- Internal structure:
  „Mr. Brandon“
- Context:
  „The new company, SafeTek, will make air bags.“

NE and chunk parsing

- POS tagging plus generic chunk parsing alone does not solve the NE problem
  - Complex modification; target structure
    - [[1 Komma 2] Mio Euro]
      CARD NN CARD NN NN
  - POS tagging and chunk parsing would construct following syntactical possible but wrong structure
    - [1 Komma] [2 Mio] [Euro]

NE and chunk parsing

- Postmodification
  - Date expression with target structure
    - Am [3. Januar 1967]
      CARD NN CARD
  - Wrong structure when generic chunk parsing
    - Am [3. Januar] [1967]
      CARD NN CARD

NE and chunk parsing

- Coordination of unit measures
  - target structure
    - [6 Euro und 50 Cents]
      CARD NN KON CARD NN
  - Generic chunk analysis
    - [6 Euro] und [50 Cents]
      CARD NN KON CARD NN

NE and chunk parsing

- Person names
  - Target structure
    - [John F. Kennedy]
      NE NE NE
  - Generic chunk parsing
    - [John F.] [Kennedy]
      NE NE NE

NE ambiguities and NE reference resolution

Norman Augustine ist im Grunde seines Herzens ein friedlicher Mensch."Ich könnte niemals auf irgend etwas schiessen", versichert der57jährige 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 abreviation lexicon, how to we recognize it? Also by taking into account NE reference resolution.

NER and answer extraction

- Often, the expected answer type of a question is an NE
  - *What was the name of the first Russian astronaut to do a spacewalk?*
    - Expected answer type is PERSON NAME
  - *Who was the first astronaut to do a spacewalk?*
    - Expected answer type either PERSON NATION or PERSON NAME
  - *Where is the Völklinger Hütte?*
    - Expected answer type is LOCATION
  - *When will be the next talk?*
    - Expected answer type is DATE

NE is an interesting problem

- Productivity of name creation requieres lexicon free recognition
- NE ambiguity require resolution methods
- Depending on the application fine-grained NE classification is needed which needs fined-grained decision making methods
- Multilinguality
  - A text might contain NE expressions from different languages (e.g., name expression)
  - Example output of IdentiFinder™

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 requiere large annotated corpora

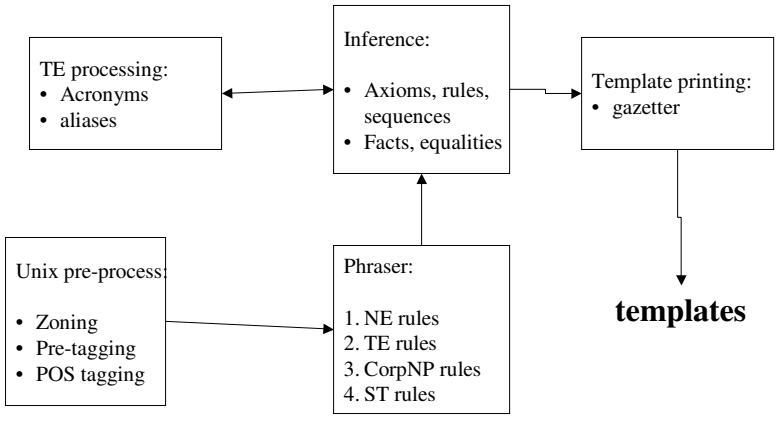
The hand-crafted approach

- 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

Example of a rule based system

- Alembic system developed at MITRE (http://www.mitre.org)
- IE core system for English (MUC participant)
- MUC-6 Alembic version
  - Transformation based rule sequence method (following Brill) on all level of processing
  - POS tagging, syntactic analysis, inference, template filling

Architecture



Phrase recognition

- Phraser processes in several steps
  1. Initial phrasing functions are applied to all of the sentences to be analyzed
     - Driven by Word lists, POS, pre-tagging
  2. Sequence of phrase-finding rules
     1. Each individual rule is applied on whole text before next rule is called
     2. If antecedents of the rules are satisfied by a phrase then action indicated by the rule is executed
     3. Possible actions: change label, grow boundary, create new phrases
     4. Next rules are sensitive to previous rules results
     5. No re-analysis of a rules action is possible (no backtracking)

Simple form of rules

- Rules can test lexems to the left/right
- Look at the lexemes in the phrase
- Tests can be POS, literal match, or generic predicates on phrase structure

(def-phraser
Label none
Left-1 phrase ttl
Label action person)

Example

... Widely anticipated: <ttl>Mr.</ttl> <none>James</none>, <num>57</num> years old, is stepping ...

(def-phraser
Label none
Left-1 phrase ttl
Label action person)

... Widely anticipated: <ttl>Mr.</ttl> <person>James</person>, <num>57</num> years old, is stepping ...

Learning rules using Brill's TBL approach

- Learning of rules for ENAMEX
- Evaluation result (6 fewer points for P&R compared to hand-crafted rules)

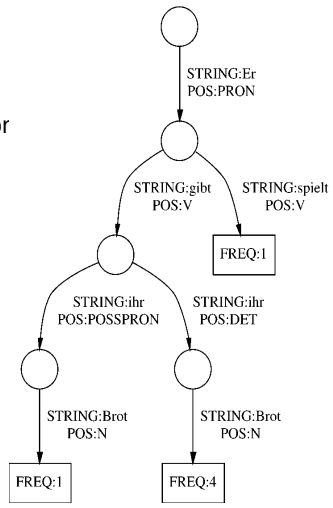
SLOT	POS	ACT	COR	PAR	INC	SPU	MIS	NON	REC	PRE	UND	OVG	ERR	SUB
Organi	454	493	392	0	28	73	34	0	86	80	7	15	26	7
Person	373	364	292	0	60	12	21	0	78	80	6	3	24	17
Locati	111	134	91	0	18	25	2	0	82	68	2	19	33	16

NE recognition of German DFKI`s SMES technology

- Based an two primary data structures
  - Weighted finite state Machines
  - Dynamic tries for lexical & morphological processing
    - recursive traversal (e.g., for compound & derivation analysis)
    - robust retrieval (e.g., shortest/longest suffix/prefix)
- Parameterizable XML-output interface
- Both tools are portable across different platforms (Unix & Windows NT)

Generic Dynamic Tries

- parameterized tree-based data structure for efficiently storing sequences of elements of any type, where each sequence is associated with an object of some other type (GDT)
- efficient **deletion** function is provided (self-organizing lexica)
- variety of complex functions relevant to linguistic processing supporting recognitior of **longest and shortest prefix/suffix** of a given sequence in the trie
- example: Trie for storing verb phrases and their frequencies, where each component of the phrase is represented as a pair <POS,STRING>



Tokenizer

- The goal of the TOKENIZER is to:
  - map sequences of consecutive characters into word-like units (tokens)
  - identify the type of each token disregarding the context
    (LOWER_CASE_WORD,
    TWO_DIGIT_NUMBER, NUMBER_PERCENT_COMPOUND, etc.)
- represented as single WFSA (easily updatable or extendable)
- generic vs. specific token classes:
  - „13:15“ (number-colon-compound)
  - „neumann@dfki.de“ (email-adress)
- complex token classes:
  - ABBREVIATION (CANDIDATE_FOR_ABBREVIATION),
  - COMPLEX_COMPOUND_FIRST_CAPITAL („*AT&T-Chief*“)
  - COMPLEX_COMPOUND_FIRST_LOWER_DASH („*d'Italia-Chefs-“)*
- overall more than 50 classes (proved to simplify processing on higher stages)
- word-segmentation: „(first,second)“ => „(„ + „first“ + „,“ + „second“ + „)“

Lexical Processor (1)

- Tasks of the LEXICAL PROCESSOR:
  - retrieval of lexical information
  - recognition of compounds
  - hyphen coordination
- sole storage: dynamic tries
- currently more than 700 000 German full-form words, generated from Morphix
- each reading represented as triple <STEM,INFLECTION,POS>

  example: „wagen“ (*to dare* vs. *a car*)

STEM: „wagen“
INFL: (GENDER: m,CASE: nom, NUMBER: sg)
(GENDER: m,CASE: akk, NUMBER: sg)
(GENDER: m,CASE: dat, NUMBER: sg)
(GENDER: m,CASE: nom, NUMBER: pl)
(GENDER: m,CASE: akk, NUMBER: pl)
(GENDER: m,CASE: dat, NUMBER: pl)
(GENDER: m,CASE: gen, NUMBER: pl)
POS: noun

STEM: „wag“
INFL: (FORM: infin)
(TENSE: pres, PERSON: anrede, NUMBER: sg)
(TENSE: pres, PERSON: anrede, NUMBER: pl)
(TENSE: pres, PERSON: 1, NUMBER: pl)
(TENSE: pres, PERSON: 3, NUMBER: pl)
(TENSE: subjunct-1, PERSON: anrede, NUMBER: sg)
(TENSE: subjunct-1, PERSON: anrede, NUMBER: pl)
(TENSE: subjunct-1, PERSON: 1, NUMBER: pl)
(TENSE: subjunct-1, PERSON: 3, NUMBER: pl)
(FORM: imp, PERSON: anrede)
POS: verb

Lexical Processor (2)

- Compound analysis
  - crucial since compounding is very productive process of German
  - realized by means of recursive trie traversal
  - often more than one decomposition: rules for validating compound morphemes

  example: „Autoradiozubehör" (*car-radio equipment*)

  (1) „Autor" + „adio" + „zubehör"
  (2) „Auto" + „radio" + „zubehör"

  example: „Weinsorten"

  (1) „Wein" + „sorten" (*wine types*)
  (2) „Weins" + „orten" *(wine places*)
- Hyphen coordination
  - in some cases conjuncts are not compounds:

  (1) „Leder-, Glas-, Holz- und Kunstoffbranche"
  leather, glass, wooden and synthetic materials industry

  (2) „An- und Verkauf" *purchase and sale*

POS-Filter (1)

- The task of POS FILTER is to filter out unplausible readings of ambiguous word forms
- large amount of German word forms are ambiguous
  - more than 20% ambiguous word forms in test corpus
  - ca. 30% of the ambiguous word forms have verb reading
- case-sensitive rules
  - „das Unternehmen" - *the enterprise* vs. „wir unternehmen" - *we undertake*
  - problems at the beginning of the sentence
- contextual filtering rules
  - example: „Sie bekannten, die bekannten Bilder gestohlen zu haben"
    They confessed they have stolen the famous pictures

    „bekannten" - *to confess* vs. *famous*
  - FILTERING RULE:

    if the previous word form is determiner and the next word form is a noun then filter out the verb reading of the current word form

POS-Filter (2)

- filtering out rare readings

  - example: „recht“ *right* vs. *to rake* (3rd person,sg)

- supplementary rules determined by Brill‘s tagger in order to achieve broader coverage
- rules may be compiled into single FST:
  - single rules expressed as nondeterministic FSTs
  - use local extension to turn the FSTs to operate globally
  - compose FSTs into single FST
  - determininze & minimize the FST

Named Entity Finder

- The task of the NAMED ENTITY FINDER is the identification of:
  - entities: organizations, persons, locations
  - temporal expressions: time, date
  - quantities: monetary values, percentages, numbers
- Identification in two steps:
  - recognition patterns expressed as WFSA are used to identify phrases containing potential candidates for named entities
  - additional constraints (depending on the type of a candidate) are used for validating the candidates and an appropriate extraction rule is applied in order to recover the named entity

    example: „von knapp neun Milliarden auf über 43 Milliarden Spanische Pesetas“
    from almost nine billions to more than 43 billions spanish pesetas

    TYPE: monetary

    SUBTYPE: monetary-prepositional-phrase

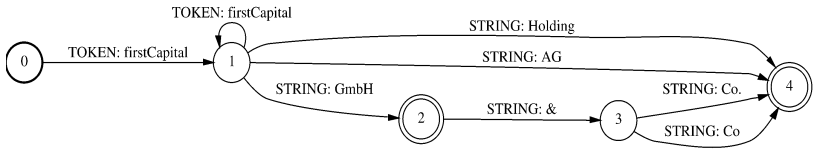
- Longest match strategy

Named Entity Finder (cont.)

- 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



ad
candidate: „Die Braun GmbH & Co.“ extracted: „Braun GmbH & Co.“

Named Entity Finder (cont.)

- Additional lexica for geographical names, first names (persons) and company names compiled as WFSA (new token classes)
- Named entities may appear without designators (companies, persons)
- Dynamic lexicon for storing named entities without designators
- Candidates for named entities, example:

  *Da flüchten sich die einen ins Ausland, wie etwa der Münchner Strickwarenhersteller **März GmbH** oder der badische Strumpffabrikant Arlington Socks, GmbH. Ab kommendem Jahr strickt **März** knapp drei Viertel seiner Produktion in Ungarn.*

- Resolution of type ambiguity using the dynamic lexicon:
  if an expression can be a person name or company name (*Martin Marietta Corp.*)
  then use type of last entry inserted into dynamic lexicon for making decision

Evaluation

- **Basis**

  corpus of German business magazine „Wirtschaftswoche“ (1,2MB, 197118 tokens)
- **Performance**

  ~10sec. (~12000wrds/sec; PentiumIII, 700MHz, 256Ram)
- **Evaluation** (20.000 tokens)

	Recall	Precision
➢ compound analysis:	98.53 %	99.29 %
➢ part-of-speech-filterung:	74.50 %	96.36 %
➢ Named entity (including NE reference resolution)	85 %	95.77 %
▪ person names:	81.27%	95.92%
▪ companies:	67.34%	96.69%
▪ locations:	75.11%	88.20%
▪ total:	73.94%	94.10%