Named Entity Recognition

CVETANA KRSTEV

UNIVERSITY OF BELGRADE, FACULTY OF PHILOLOGY

DEPARTMENT OF LIBRARY AND INFORMATION SCIENCES

Outline of my talk

Defining „Named Entities“

Named Entity tagging

- Different schemas

Named Entity Recognition Methods

An Example

Defining „Named Entities“

Named entity recognition

Named entity recognizers identify proper names in documents, and may also classify these proper names as to whether they designate people, places, companies, organizations, and the like.

In the sentence:

- **Italy**'s business world was rocked by the announcement **last Thursday** that **Mr. Verdi** would leave his job as vice-president of **Music Masters of Milan, Inc** to become operations director of **Arthur Andersen**.

'Italy' would be identified as a place, 'last Thursday' as a date, 'Verdi' as a person, 'Music Masters of Milan, Inc' and 'Arthur Andersen' as companies.

Some would consider recognition of 'Milan' as a place, and identifying 'Arthur Andersen' as a person as an error in this context.

Named Entity Definition

In the expression **named entity**, the word ***named*** restricts entities to those for which one or many rigid designators stands for the referent.

Example:

- **the automotive company created by Henry Ford in 1903**
- is referred to as **Ford** or **Ford Motor Company**

Rigid designators include proper names as well as terms for certain biological species and substances.

Rigid Designator

What is a rigid designator?

As defined by **Saul Kripke** in *Naming and Necessity*. Cambridge, Mass.: Harvard University Press

According to Kripke a name refers to an object by virtue of a causal connection with the object as mediated through communities of speakers.

He also points out that **proper names**, in contrast to most descriptions, are **rigid designators**. That is, a proper name refers to the named object in every possible world in which the object exists, while most descriptions designate different objects in different possible worlds.

For example, 'Nixon' refers to the same person in every possible world in which Nixon exists, while 'the person who won the United States presidential election of 1968' could refer to Nixon, Humphrey, or others in different possible worlds.

Do all NEs conform to this definition?

Temporal expressions and some numerical expressions (i.e., money, percentages, etc.) may also be considered as named entities in the context of the NER task.

Some instances of these types are good examples of rigid designators, for example:

- **the year 2001**
- **2001. godina**

Because in both cases they refer to the ***2001st year of the Gregorian calendar***.

Temporal expressions in some other cases do not have rigid designators, for instance:

- **in June** – we don't know whether it refers to: **past June**, **this June**, **June 2020**...

Named Entity Schemas

The purpose of NE schemas

To standardize tagging by and for applications (MUC, ENE)

To standerdize tagging by and for people (TEI)

Message Understanding Conferences

They were initiated by **DARPA** (**Defense Advanced Research Projects Agency**) to encourage development of new and better methods for extracting information from unstructured texts.

Seven annual conferences were held between 1987 and 1997.

These were not traditional scientific conferences but rather competitions in which research teams competed with their information extraction systems.

- All systems always used the same texts from the particular domain. For instance, the task of MUC-3 and MUC-4 was event extraction and the domain were terrorist attacks as reported in newswires.
- The output of all systems was standardized.

Named entities and MUC

MUC-6 (1995) introduced Named Entity extraction as a component task, i.e., the finding of proper names of people, companies, places, etc. in free text, but also continued event extraction of management changes in the news

In 1998, MUC-7 showed that Named Entity extraction from English language newswire articles was more or less a solved problem. The best MUC-7 programs scored about $F = 93\%$, compared to an estimated human performance of about $F = 97\%$.

The **TIPSTER** program of which **MUC** was a part was wound up after MUC-7.

Named entities and coreferences

MUC defined a coreference task as linking together multiple expressions that refer to a given entity.

In the context of information extraction, the role of coreference annotation is to ensure that information associated with multiple mentions of an entity can be collected together.

For instance,

- **<coref id='100'>**International Business Machines **</coref>**
- **<coref id='101' type='ident' ref='100'>**IBM**</coref>**

The acronym **IBM** refers to the identical notion as the phrase **International Business Machines**.

Markup introduced by MUC-6 for named entites

All systems that participated in MUC competitions had to produce texts annotated with **SGML** tags (**SGML** was an **XML** predecesor) that was a valid SGML (e.g. XML) dokument, that is, conforming to the prescribed DTD.

- **Named entites proper – ENAMAX**
- **Temporal expressions – TIMEX**
- **Number expressions – NUMEX**

ENAMAX element

This subtask is limited to proper names, acronyms, and perhaps miscellaneous other unique identifiers, which are categorized via the **TYPE** attribute as follows:

- **ORGANIZATION**: named corporate, governmental, or other organizational entity;
- **PERSON**: named person or family
- **LOCATION**: name of politically or geographically defined location (cities, provinces, countries, international regions, bodies of water, mountains, etc.). This tag is not intended for addresses, names of streets, adjectives derived from location names, etc.

Some examples of **ENAMAX** tags/1

Organizations:

- **<ENAMEX TYPE="ORGANIZATION">European Community</ENAMEX>**
- **<ENAMEX TYPE="ORGANIZATION">Labor Party</ENAMEX>**
- **<ENAMEX TYPE="ORGANIZATION">Finger Lakes Area Hospital Corp.</ENAMEX>**

Some examples of **ENAMAX** tags/2

Titles such as „Mr.“ and role names such as „President“ are ***not*** considered part of a person name. However, appositives such as „Jr.“ ***are*** considered part of a person name.:

- Mr. **<ENAMEX TYPE="PERSON">Harry Schearer</ENAMEX>**
- Secretary **<ENAMEX TYPE="PERSON">Robert Mosbacher</ENAMEX>**
- **<ENAMEX TYPE="PERSON">John Doe, Jr.</PERSON>**

The example of a familiy name:

- **the <ENAMEX TYPE="PERSON">Kennedy</ENAMEX> family**

Some examples of **ENAMAX** tags/3

Country name is a part of a name of an organization:

- **<ENAMEX TYPE="ORGANIZATION">Hyundai of Korea, Inc.</ENAMEX>**

Country name is not a part of a name of an orgnaization

- **<ENAMEX TYPE="ORGANIZATION">Hyundai, Inc.</ENAMEX> of <ENAMEX TYPE="LOCATION">Korea</ENAMEX>**

City name is not a part of a name of an university

- **<ENAMEX TYPE="ORGANIZATION">University of California</ENAMEX> in <ENAMEX TYPE="LOCATION">Los Angeles</ENAMEX>**

Compound expressions in which place names are separated by a comma are to be tagged as separate instances of **LOCATION**

- **<ENAMEX TYPE="LOCATION">Kaohsiung</ENAMEX>, <ENAMEX TYPE="LOCATION">Taiwan</ENAMEX>**

TIMEX element

Only „absolute“ time expressions are to be tagged. To be considered an absolute time expression, the expression must indicate a specific segment of time.

The tagged tokens are categorized via the **TYPE** attribute:

- **DATE**: expressions that fully or partially designate a calendar date;
- **TIME**: expressions that fully or partially designate a time of a day;

Some examples of TIMEX tags

Time

- <TIMEX TYPE="TIME">twelve o'clock noon</TIMEX>
- <TIMEX TYPE="TIME">5 p.m. EST</TIMEX>

Date

- <TIMEX TYPE="DATE">January 1990</TIMEX>
- <TIMEX TYPE="DATE">fiscal 1989</TIMEX>
- the <TIMEX TYPE="DATE">autumn</TIMEX> report (?)
- <TIMEX TYPE="DATE">third quarter of 1991</TIMEX>
- <TIMEX TYPE="DATE">the fourth quarter ended Sept. 30</TIMEX>

NUMEX element

This subtask is for two useful types of numeric expressions, monetary expressions and percentages. The numbers may be expressed in either numeric or alphabetic form.

The task covers the complete expression, which is categorized via the **TYPE** attribute:

- **MONEY**: monetary expression
- **PERCENT**: percentage

Some examples of **NUMEX** tag

monetary expressions:

- **<NUMEX TYPE="MONEY">20 million New Pesos</NUMEX>**
- **<NUMEX TYPE="MONEY">$42.1 million</NUMEX>**
- **<NUMEX TYPE="MONEY">million-dollar</NUMEX> conferences**

percentage

- **<NUMEX TYPE="PERCENT">15 pct</NUMEX>**

The enhancement of the named entity set

The named entity set included in MUC-6 was very modest and those included were categorized in very broad groups.

In Named Entity Recognition and Classification (NERC) field researchers suggested the enhancement of this set and a finer categorization, for instance:

- The category **person** can be further subcategorized as politician, entertainer, etc.
- The same category **person** can be used for recognition of other interesting entities, such as names of diseases and medicaments, e.g. ***Parkinson disease***, name of dishes ***Tournedos Rossini***.

The Named Entity Hierarchy – ENE project

The extended Named Entity Hierarchy (ENE) for newspaper texts was proposed by **Satoshi Sekine** from the New York University and **Chikashi Nobata** from Communications Research Laboratory, Japan

This hierarchy has approximately 200 categories. At the top are still three categories: **NAME**, **TIME_TOP** and **NUMEX**.

Using this hierarchy they developed dictionaries and automatic taggers for NEs in Japanese and English.

These methods are very useful for applications in information retrieval, question answering, term extraction, etc.

The example of the new category - product

TOP

>NAME

>>**PRODUCT** (a product is a type of a 'name')

>>>PRODUCT_OTHER (for products that cannot be further categorized)

>>>***VEHICLE*** (a vehicle is a type of a 'product')

>>>>VEHICLE_OTHER CAR TRAIN AIRCRAFT SPACESHIP SHIP

>>>***FOOD CLOTHS DRUG WEAPON STOCK AWARD THEORY RULE SERVICE CHARACTER METHOD_SYSTEM DOCTRINE CULTURE RELIGION LANGUAGE PLAN ACADEMIC CLASS SPORTS_OFFENCE***

>>>***ART***

>>>>ART_OTHER PICTURE BROADCAST_PROGRAM MOVIE SHOW MUSIC BOOK

>>>***PRINTING***

>>>>PRINTING_OTHER NESPAPER MAGAZINE

An example of the refinement of existing categories

>**TIME_TOP**

>>TIME_TOP_OTHER

>>>**TIMEX**

>>>>TIMEX_OTHER TIME DATE DAY_OF_WEEK ERA

>>>**PERIODX**

>>>>PERIODX_OTHER TIME_PERIOD DATE_PERIOD WEEK_PERIOD MONTH_PERIOD YEAR_PERIOD

Other newly introduced categories

Category name (**NAME**)

- **FACILITY** (stadium, theater, etc.)
- **EVENT**
- **NATURAL_OBJECT** (living beings, minerals)
- **TITLE**

Number expressions (**NUMEX**)

- **MEASUREMENT**
- **COUNTX**
- **ORDINAL_NUMBER**

Named Entity Categories and TEI

One chapter of TEI (Text Encoding Initiative) guidelines is dedicated to named entities:

- **P5: Guidelines for Electronic Text Encoding and Interchange**
- Chapter 13: **Names, Dates, People, and Places**

Elements and their attributes are described in this chapter that can be used when a special TEI module is included **namesdates** – without it only basic elements can be used, for instance for names those are **name** and **rs**.

Person names in TEI

<persName>

- **<surname>**
- **<forename>**
- **<roleName>**
- **<addName>**
- **<nameLink>**
- **<genName>**

Examples:

```
<persName key="DUDO1">
  <roleName type="honorific" full="abb">Mme
  </roleName>
  <nameLink>de la</nameLink>
  <surname>Rochefoucault</surname>
</persName>
<persName>
  <forename>Charles</forename>
  <genName>II</genName>
</persName>
```

Geopolitical names in TEI

<placeName>

- **<district>**
- **<settlment>**
- **<region>**
- **<country>**
- **<bloc>**

Examples:

```
<placeName key="LSEA1">
  <country type="nation">Laos</country>,
  <bloc type="sub-continent">Southeast Asia</bloc>
</placeName>
<placeName>
  <settlement type="city">Rochester</settlement>,
  <region type="state">New York</region>
</placeName>
```

Geographic names in TEI

<geogName>

- **<geogFeat>**

Examples:

```
<geogName key="MIRI1" type="river">
<name>Mississippi</name>
<geogFeat>River</geogFeat>
</geogName>
<geogName>
<geogFeat>Mount</geogFeat>
<name>Sinai</name>
</geogName>
```

Organization names in TEI

<orgName> - Examples:

- **About a year back, a question of considerable interest was agitated in the <orgName key="PAS1" type="voluntary"> <placeName key="PEN">Pennsyla.</placeName> Abolition Society</orgName>....**
- **A spokesman from <orgName type="regional"> <orgName type="acronym">IBM</orgName> <country type="acronym">UK</country> </orgName> said ...**

Dates in TEI

<date> - Examples:

- **<date when="1980-02">early February 1980</date>**
- **He was born on <date calendar="Gregorian">Feb. 22, 1732</date> (<date calendar="Julian" when="1732-02-22">Feb. 11, 1731/32, O.S.</date>).**
- **In New York, <date type="occasion" when="--01-01">New Years Day</date> is the quietest of holidays, <date when="--07-04" type="occasion">Independence Day</date> the most turbulent.**

Time in TEI

<time> - Examples:

- As he sat smiling, the quarter struck —

  <time when="11:45:00">the quarter to twelve</time>

- The train leaves for Boston at

  <time type="twentyfourHour" when="13:45:00">a quarter of two</time>

- I reached the station <time when="14:15:00">

  <time dur="PT30M0S">precisely half an hour</time>

  <offset>after</offset>

  <time when="13:45:00" type="occasion">the departure of the afternoon train to Boston</time>

  </time>

Read more

1. Examples and explanations for MUC schema are taken from Ralph Grishman: Named Entity Task Definition, 1995, **http://cs.nyu.edu/faculty/grishman/NEtask20.book_1.html**
2. Text Encoding Guidelines, **http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ND.html**
3. Sekine, S. and C.Nobata „Definition, dictionaries and tagger for Extended Named Entity Hierarchy“, LREC 2004.
4. Sekine, Satoshi. "Extended Named Entity Ontology with Attribute Information." LREC. 2008.

Named Entity Recognition Methods

Problems with NER

Many referring expressions are proper names and may therefore exhibit initial capital letters in English (and many other European languages), e.g., **John Smith**, **Thomson Corporation** and **Los Angeles**.

The presence of an initial capital does not guarantee that one is dealing with part of a name, since initial capitalization is also used:

- at the start of sentences,
- Variables in mathematics, chemical symbols, **X-rays**,...
- Acronyms that are not named entites (**FC** – for football club)
  - Acronyms in short messages: **OMG** (Oh, my God), etc.

Also, for some named entities no initial capital letter is used, e.g. **eBay**.

Use of gazetteers

NER task can be simplified by using lists of people, places and companies, but that cannot be enough.

Using a directory or gazetteer doesn't necessarily help a program to decide whether **Philip Morris** refers to a person or a company.

New companies, products, etc. come into being on a daily basis, and they also change names, are referred to by some informal or shortened name, e.g. a Serbian machine factory **Ivo Lola Ribar,** usually referred to as **Lola**.

Gazetteers cannot help with temporal and number expressions.

Referring expressions and relations

Directories and gazetteers might help with proper names, but not other referring expressions.

Some are

- definite descriptions, e.g., **the famous inventor**,
- others are pronouns, such as **he**, **she**, or **it**.

Most software packages for NER concentrate upon identifying proper names that refer to people, places and companies.

They may also try and find relationships between entities, e.g., **Bill Gates, President of Microsoft Corporation** will yield the person **Bill Gates** standing in a ***President*** relationship to the company **Microsoft**.

Heuristic approaches to NER

In the 7th MUC conference, there was a track devoted to named entity recognition, with data collections and test conditions being set up along the lines of earlier conferences.

The best MUC-7 system came from Edinburgh University, Language Technology Group which employed a variety of methods, combining lists, rules, and probabilistic techniques, applied in a particular order.

Their system achieved F-score 93.39, with balanced precision and recall.

They outperformed other systems particularly in recognition of **organization** named entity.

Steps applied in LTG NER system (1)

1. The application of high-confidence heuristic rules. This rules rely heavily upon syntactic cues in the surrounding context.
   - **John Smith, director**, we know that **John Smith** refers to a person, because a string of capitalized words followed by a title or profession indicates the name of a person with high reliability.
   - Similar rules can be written to recognize names of companies or organizations in expressions such as **president of Microsoft Corporation**.
2. The system also uses lists of names, locations, etc., but at this stage it only checks to see if the context of a possible entity supports suggestions from the list. For example, a place name like **Washington** can be also surname or the name of an organization. Only in a suggestive context, like **in the Washington area**, would it be classified as a location.

Steps applied in LTG NER system (2)

3. All named entities already identified in the document are collected and partial orders of the composing words are created. For instance, if **Lockheed Martin Production** has already been tagged as an organization, because it occurred in the list of organization names and occurred in a context suggestive of organizationsthen, all instances of **Lockheed Martin Production**, **Lockheed Martin**, **Lockheed Production**, **Martin Production**, **Lockheed** and **Martin** will be marked as possible organizations. The annotated stream is then fed to a trained statistical model that tries to resolve some of the suggestions.
4. The system again applies its rules, but with much more relaxed contextual constraints. Organizations and locations from the lists available to the system are marked in the text, without checking the context in which they occur. If a string like **Philip Morris** has not been already tagged as an organization, then the name grammar will tag it as a person without further checking of the context.

Steps applied in LTG NER system (3)

5. The system then performs another partial match to label short forms of personal names, such as **White** when **James White** has already been recognized as a person, and to label company names, such as **Hughes** when **Hughes Communications** has already been identified as an organization.
6. Because titles of documents are often in capital letters, they provide little guidance for the recognition of names. In the final stage of processing, entities in the title are marked up, by matching or partially matching the entities found in the text, and checking against a statistical model trained on document titles. For example, in the headline **MURDOCK SATELLITE EXPLODES ON TAKE-OFF**, **Murdoch** will be tagged as a person because it partially matches **Rupert Murdoch** elsewhere in the text.

Statistical approach to NER

An alternate approach to NER is to write a program that learns how to recognize names.

The method often used is called **Hidden Markov Models**.

Some key work in this area derives from the **Nymble** system, which participated in MUC-6 and MUC-7, and has since then transformed into the more highly developed **Identifinder** system.

This approach means that it is necessary to learn to decide, for each word in the text, whether or not it is part of a name. Also, it is important to decide what kind of name has been found, that is, to which name class a word belongs to. The system uses 7 name classes and, for convenience, a name class NOT-A-NAME is also included.

Word features

As with the heuristic approach, it is necessary to identify mutually exclusive features of words that provide clues as to what kinds of words they are.

The **Nymble** system used 14 categories.

These word features are not informative enough, in themselves, to identify names, or parts of names, reliably on a word-by-word basis. However, they can be leveraged, in conjunction with information about word position and adjacency, to provide better estimates of name class.

This approach is based on the commonly used bigram language model, in which a word's probability of occurrence is based on the previous word. The probability of a sequence of words $< w_0, \ldots, w_n >$ is then computed by the product

$$P(w_1 \mid w_0)\, P(w_2 \mid w_1) \ldots P(w_n \mid w_{n-1})$$

with a **START-OF-SENTENCE** word being used to compute the probability of w_1.

Feature set in Nymble system		
Word Feature	**Example**	**Explanation**
twoDigitNum	90	Two-digit year
fourDigitNum	1990	Four-digit year
containsDigitAndAlpha	A8-67	Product code
containsDigitAndDash	09-96	Date
containsDigitAndSlash	11/9/98	Date
containsDigitAndComma	1,000	Amount
containsDigitAndPeriod	1.00	Amount
otherNum	12345	Any other number
allCaps	BBC	Organization
capPeriod	P.	Personal name initial
firstWord	The	Capitalized word that is the first word in a sentence
initCap	Sally	Capitalized word in midsentence
Lowercase	tree	Uncapitalized word
other	.net	Punctuation, or any other word not covered above

Satistical model (1)

One important source of information is the name class assigned to the previous word in the sentence. Another is the preceding word itself. Thus one component of assigning a name class, NC_0, to the current word, w_0, is given by the following probability:

$$P(NC_0 \mid NC_{-1}, w_{-1})$$

where NC_{-1} is the name class of the previous word, w_{-1}.

Another component looks at the probability of generating the current word and its associated feature, given the name class assigned to it and the name class of the previous word, i.e.,

$$P(< w_0, f_0 > \mid NC_0, NC_{-1})$$

where $< w_0, f_0 >$ stands for the current word-feature pairing.

Satistical model (2)

These probabilities are combined into the following model for generating **THE FIRST WORD** of a name class:

$$P(NC_0 \mid NC_{-1}, w_{-1}) \cdot P(\langle w_0, f_0 \rangle \mid NC_0, NC_{-1})$$

The model for generating **ALL BUT THE FIRST WORD** of a name class uses the word feature pair of the previous word, and current name class:

$$P(\langle w_0, f_0 \rangle \mid \langle w_{-1}, f_{-1} \rangle, NC_0)$$

There is also a distinguished **+end+** word so that the probability may be computed for any current word to be the final word of its name-class.

$$P(\langle \text{+end+}, other \rangle \mid \langle w, f \rangle_{final}, NC_0)$$

Example

For example, consider the sentence **Mr. Smith sleeps.**, in which **Smith** is in the **PERSON** name class, and the other words are not names. To compute the probability of this sequence of words, we need to include the following probabilities.

P(NOT-A-NAME | START-OF-SENTENCE, "+end+") *
P("Mr." | NOT-A-NAME, START-OF-SENTENCE) *
P("+end+" | "Mr.", NOT-A-NAME) *
P(PERSON | NOT-A-NAME, "Mr.") *
P("Smith" | PERSON, NOT-A-NAME) *
P("+end+" | "Smith", NOT-A-NAME) *
P(NOT-A-NAME | PERSON, "Smith") *
P("sleeps" | NOT-A-NAME, PERSON) *
P("." | "sleeps", NOT-A-NAME) *
P("+end+" | ".", NOT-A-NAME) *
P(END-OF-SENTENCE | NOT-A-NAME, ".")

the probability of **Mr.** starting the sentence

the occurrence of **Smith** as a person, if is preceded by a no-name **Mr.**

the occurrence of **sleeps** as a non-name, if it is preceded by a person name, **Smith**.

the occurrence of „." as an end of a sentence, if it is preceded by a non-name, **sleep**.

Calculation of probabilities

The needed probabilities are estimated from corpus counts. For example,

$$P(NC_0 \mid NC_{-1}, w_{-1}) = c(NC_0, NC_{-1}, w_{-1}) \,/\, P(NC_{-1}, w_{-1})$$

where $c(NC_0, NC_{-1}, w_{-1})$ stands for the number of times that a word of name class NC_0 follows word w_{-1} of name class NC_{-1}, and $P(NC_{-1}, w_{-1})$ stands for the total number of occurrences of word w_{-1} with name class NC_{-1}.

Read More

1. Peter Jackson and Isabelle Moulinier, *Natural Language Processing for Online Applications – Text Retrieval, Extraction and Categorization*, 2nd, John Benjamins Publishing Co, 2007.
2. Mikheev, Andrei, Claire Grover, and Marc Moens. "Description of the LTG system used for MUC-7." Proceedings of 7th Message Understanding Conference (MUC-7). Fairfax, VA, 1998.
3. Bikel, Daniel M., et al. "Nymble: a high-performance learning name-finder." Proceedings of the fifth conference on Applied natural language processing. Association for Computational Linguistics, 1997.
4. Bikel, Daniel M., Richard Schwartz, and Ralph M. Weischedel. "An algorithm that learns what's in a name." Machine learning 34.1-3 (1999): 211-231.

One Example

NER system for Serbian

Entities that are tagged belong to classes:

- **Person names** (full names and distinguished person names) their titles, roles and functions, if present, preceding or following them;
- **Geoplolitical names** – countries and settlments – **geographic names** – water bodies and oronyms.
- **Organization names** – including names of political parties.
- **Number expressions – monetary, measurements, count, percentage**
- **Time expressions – dates, times of day, periods and frequencies,** absolute and relative

General resources used for the Serbian NER

Comprehensive morphological e-dictionaries of Serbian in DELA/DELAF format:

- simple words,
- Multi-word names;

including:

- general lexica,
- geographic names,
- personal names,
- encyclopedic knowledge (in development).

Dictionary entries are provided with elaborate semantic markers.

Examples of Dictionary Entries

Geographic names:

- **Dunav,N+NProp+Top+Hyd** (Δούναβης is a proper name, geographic notion, hydronym)
- **Egejsko more,N+NProp+Top+Hyd** (Αιγαίο Πέλαγος)

Geopolitical names:

- **Solun,N+NProp+Top+Gr** (Θεσσαλονίκη – a proper name, city)
- **Helenska republika,N+NProp+Top+Dr** (Ελληνική Δημοκρατί – a proper name, country)

Organizations:

- **Atinska novinska agencija,N+NProp+Org+Acr=ANA** (Αθηναϊκό Πρακτορείο Ειδήσεων)

Person names:

- **Venizelos,N+NProp+Hum+Last+Cel** (Βενιζέλος – a last name of a famous person)
- **Riga od Fere,N+NProp+Hum+Last+Cel** (Ρήγας Φεραίος – a full name of a famous person)

The general approach – rule-based supported by lexical resources

Use of dictionaries

Use of local grammars to specify the context

- For rejecting false recognitions
- For accepting false rejections

A task: recognition and tagging of **hydronyms** (water bodies) in Serbian newspaper texts.

Problems: hydronyms are ambiguous with:

- other geographic names: ***Bosna*** – a river and a region.
- personal names: ***Una*** – a river and a feminine name, ***Sava*** – a river and a masculine name
- Common nouns: ***Kupa*** – a river, and but als a form of a noun ***kup***, a verb ***kupati***...

The first solution

We use as a text a small collection of news dealing with recent floods in Serbia in 2014 named ***Poplave*** (~10.000 simple words)

For retrieving names of water bodies we use a pattern:

- **<N+NProp+Top+Hyd>**

All names of water bodies (recorded in e-dictionaries) but also a number of false recognitions:

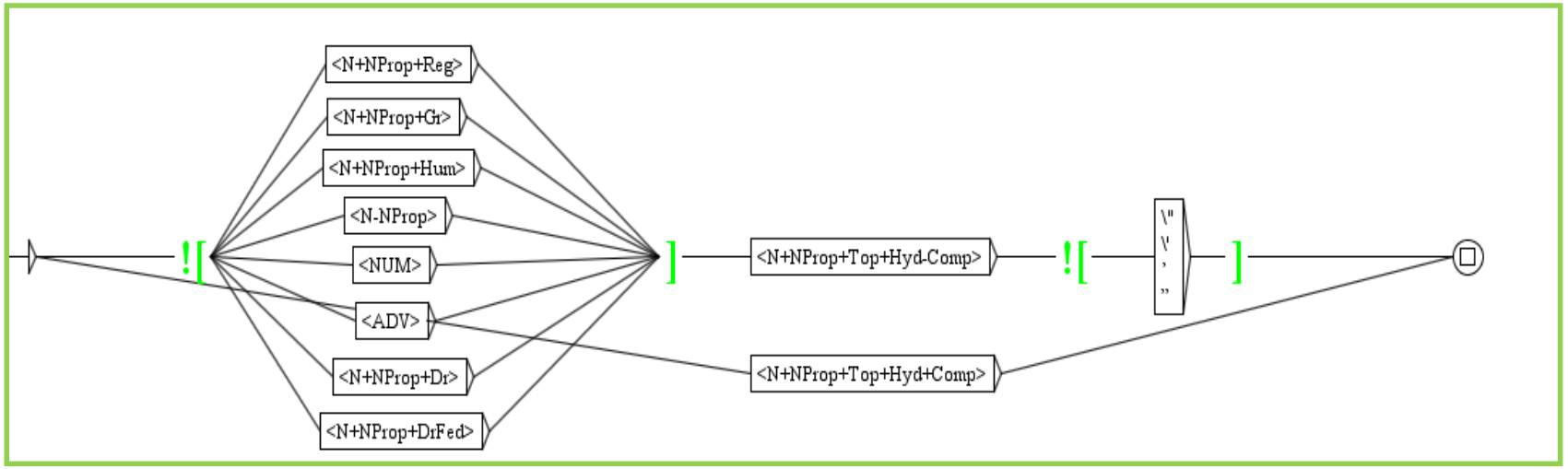
- **Oko** – a preposition 'around' and a form of a name **Oka** (Ποταμός Οκά)

89 matches / 7 false matches

The first improvement

We use a local grammar in the form of a finite-state transducer that:

- We take into consideration only simple word hydronym names that are not ambiguous with other proper or common names.
- We recognize MWU hydronym names in dictionaries (usually they are not ambiguous)



The first improvement – recall is falling

This graph retrieves some names of water bodies but also rejects some correct recognitions.

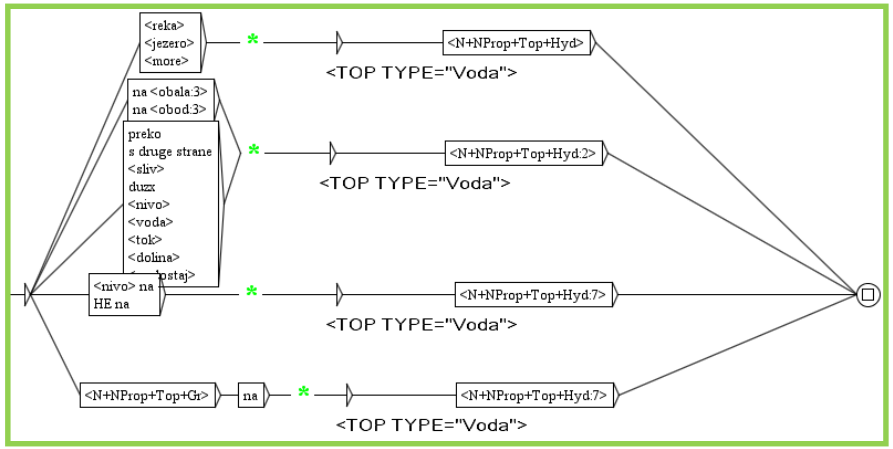
- 40 matches in a collection ***Poplave***
- No false matches
- differences from a previous recognition
- 42 false rejections

The second improvement

We try to retrieve some of falsely rejected matches.

This graph matches names of water bodies if they have a “right” left context, like

- river, lake...
- On the bank of...
- Hydropower on...



The second improvement – recall is rising

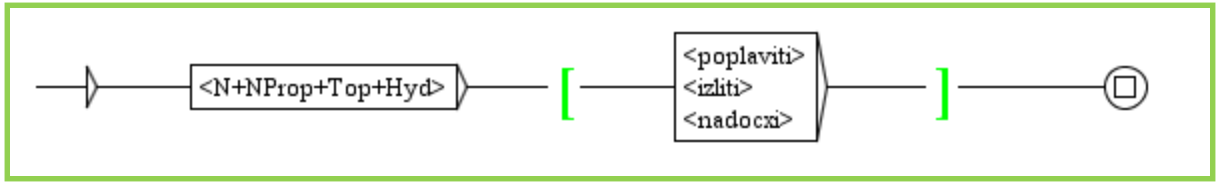
- 54 matches in a collection ***Poplave***
- differences from a previous recognition
- No false matches

The third improvement

We try to retrieve some more of falsely rejected matches.

This graph matches names of water bodies if they have a "right" right context.

- Example: ***Sava*** *je poplavila vikend naselje...* (Sava has flooded a weekend settlment...)
- 61 matches in a collection **`Poplave`**
- differences from a previous recognition

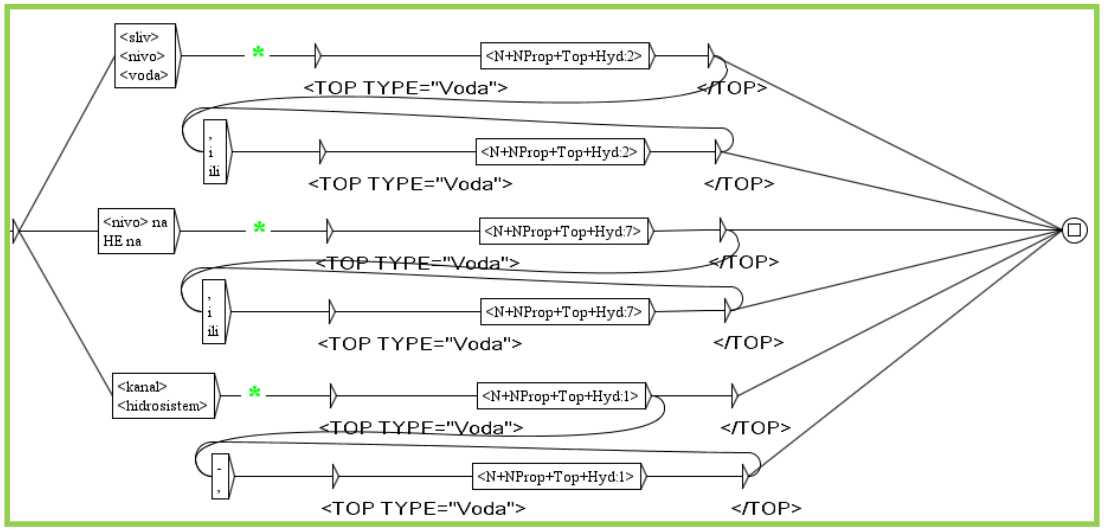


The forth improvement

We try to retrieve some more of falsely rejected matches.

This graph matches names of water bodies if they appear in a sort of a list of water body names with the ”right” left context, or if they appear together with already recognized names.

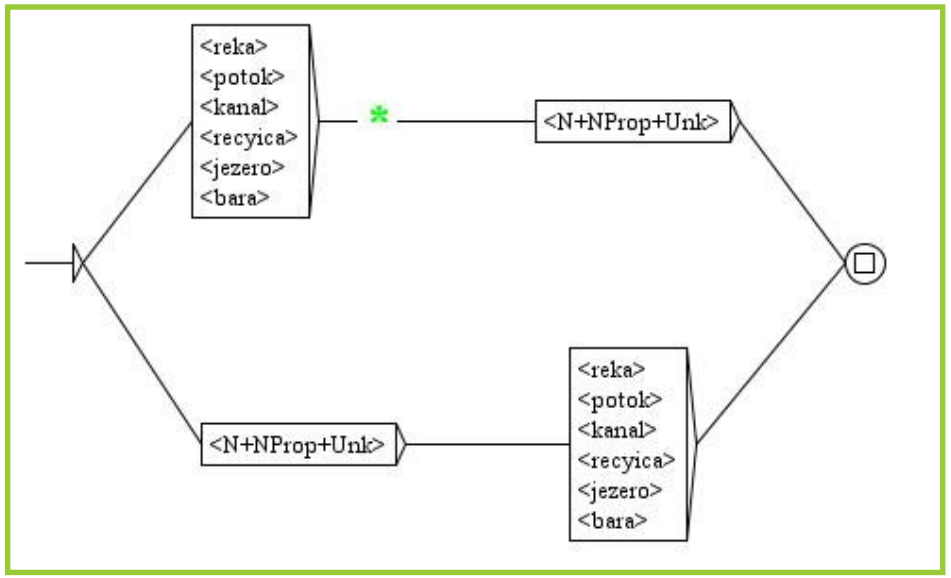
- Example: *basin, hydrosystem, etc.*
- 59 matches in a collection ***Poplave*** / 65 correct hydronym names / no false recognitions
- differences from a previous recognition



The fifth improvement

We try to retrieve some more entries – even those uppercase unknown words but with an obligatory key word following or preceding it

- 76 matches in a collection **`Poplave`** / no false recognitions / 82 correct hydronym names
- differences from a previous recognition
- Example: *Tulovska reka,* (reka) *Lugomir*



Read more

Cvetana Krstev, Ivan Obradović, Miloš Utvić, Duško Vitas, “A system for named entity recognition based on local grammars”, J Logic Computation 24(2), pp. 473-489, 2014, Oxford Journals, doi:10.1093/logcom/exs079, first published online February 19, 2013

Paumier, S. "Unitex 3.1 User Manual (2016).,, http://www-igm.univ-mlv.fr/~unitex/