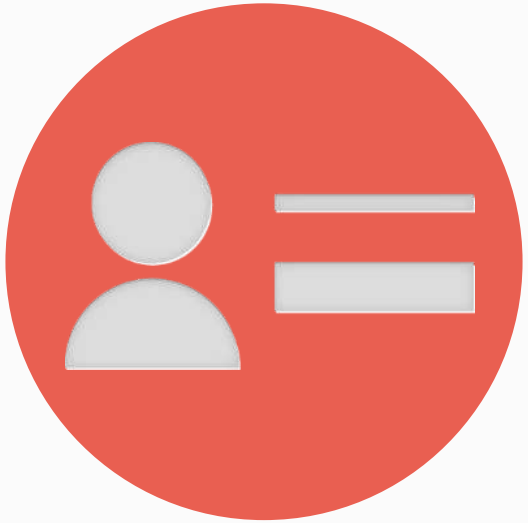
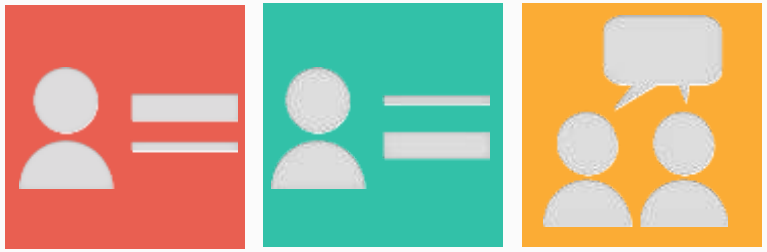


Inferring standard name form,

gender and nobility from historical texts using stable model semantics

Vitek, Darko (Dep. of History; UNIZG); Lauc, Davor (Dep. of Logic, UNIZG)

GCDH 30th May 2016



Presentation plan

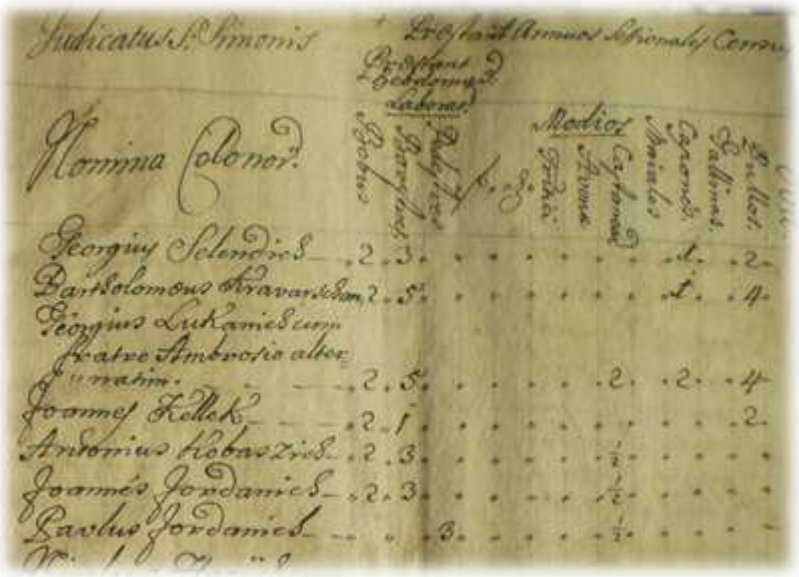
(first part)

} 16:00 -16:45

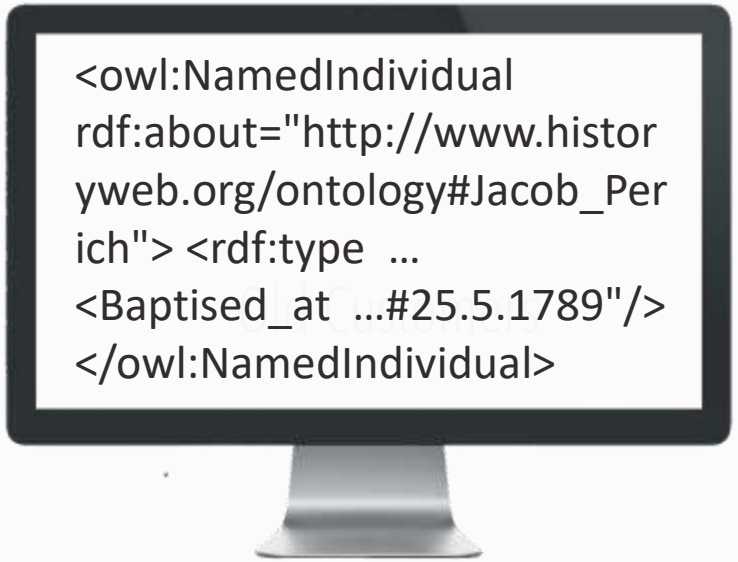
- BIG PICTURE
- Characteristics of Historical sources: In general, In particular
- "SMALL" PROBLEM: parsing and understanding proper names, extracting information from names
- State of the Art: Machine learning and statistical NLP to the rescue
- Rule based approach: Using rules to infer standard name form
- Our research – Inferring standard name form using default model semantics
- Lessons learned, Future research

Big picture

From images of the historical sources to the linked data representation



Historical sources are unstructured information, concealed in hard to decode formats, that are laden with ambiguity. Analyzing such sources is an extremely tedious exercise, that is prone to error. They are generally unfitting for computational analysis.



Structured information are much easier to understand and it is possible to analyze them using computational models. This is particularly case with semantic web representation (linked data) where even semantic ambiguity is reduced. This facilitate advanced computation analysis and inferring knowledge from information.

Serial historical sources

Features of the serial sources

Serial sources

Serial sources are historical sources characterised by systematic repetition *of structure*. For example; tax lists, censuses, parish books (liber baptizatorum, ...) etc.

Data

Due to its repetitive character, serial sources usually contain plenitude of data. This voluminous requires a lot of time and other resources to process and understand.

Classic historical method

Processing data in a serial source, for example 18th century tax list that consist of a 50 thousands records, is a substantial historiographical problem.

Computing

Using computing in the Croatian historiography is nowadays customary mostly for statistical purposes. Though, statistical and more advanced data analysis must be preceded by converting unstructured data into a more suitable formats.

Characteristics of the serial sources in Croatia

Middle age

From 11th century Croatia is a part of the Hungarian empire, which is characterised by undeveloped state institutions.

Characteristics of the sources

There are fairly limited number of sources, mostly collected into the diplomatic collections. The language of the sources was almost exclusively Latin.

Modern age

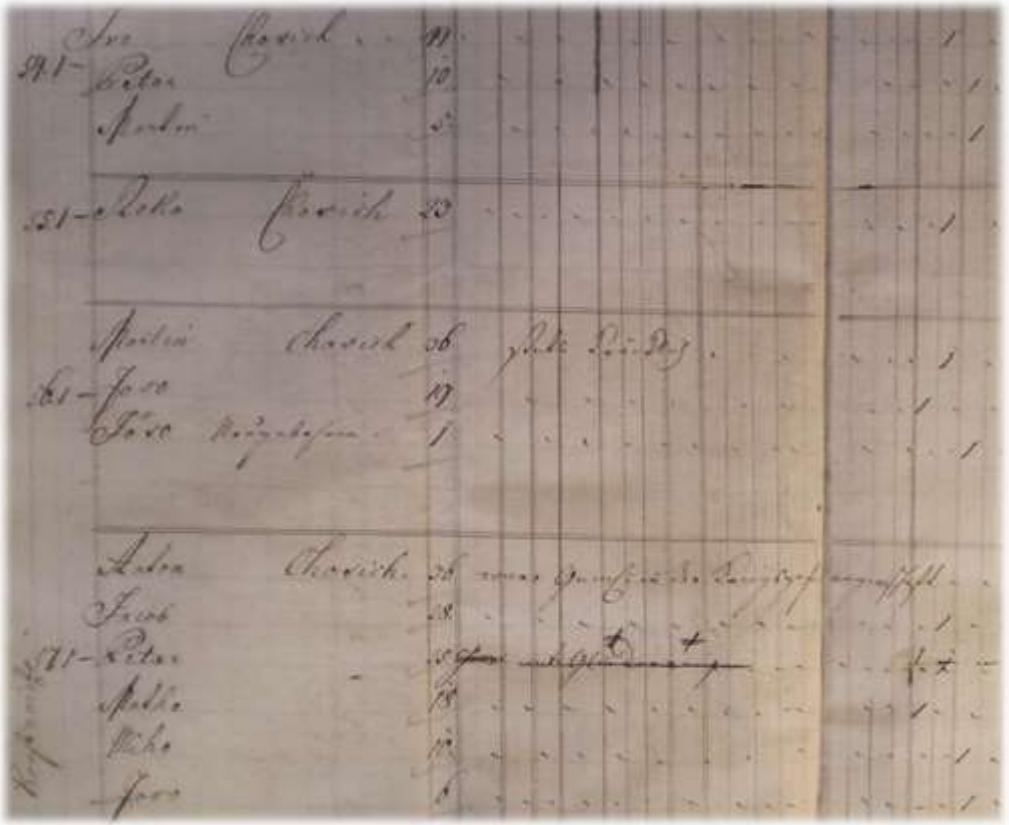
15th and 16th century Ottoman Empire conquest put an end to the middle age Hungarian state. The Kingdom of Croatia was an administrative division that existed between 1527 and 1868 within the Habsburg Monarchy. This period was characterised by unification processes. Modernization of Croatia is accompanied by development of numerous state's institutions.

Characteristics of the sources

The multiplicity of historical sources were produced during this period. The German language is starting to dominate. As the result of development of the state administration, especially in 18th and 19th century, huge number of historical sources were created. For the many of them, historiographic analysis is yet to be done.

Our test case: Zagreb's 1857 census

Hisotorical serial sources in Croatia



Census in the year 1857

This census was the first modern census in Croatia. In the area within the borders of the modern Croatia, more than 1.200.000 people were enlisted in the census.

Content of the census

The census was structured by the households. Because of that, every census record contains: address of the household, the owner of the household, all the suites in the household, and personal data about inhabitants.

Data about inhabitants

In the census there is separate data about every inhabitant. Data includes: name expression (first and last name, title, ...), data of birth, religious conviction, occupation/source of income, fatherland status, place of origin, whereabouts in the census time, and a note.

Standard NLP Pipeline

Common approach to serial sources analysis

Digitalisation: Scanning/OCR or Transcription, Dirty text normalisation

The first phase of converting unstructured historical data into structured one is either scanning/OCRing of the sources of transcription of them. Both approaches are laden with challenges.

Both OCR & transcription of handwritten text requires recognition of indecipherable handwritten text, and is prone to errors.

Tokenization & Segmentation

The second phase is usually breaking machine readable text into the smallest processing units (tokens/tokens) and larger segments (sentences / records / paragraphs).

Even this simple phase is far from trivial and can impact accuracy of next phases. Available tokenizers and sentence segmentation models perform satisfactory on standard modern text; but no so-well on historical sources.

Named entity recognition, Entity resolution

Named entity recognition phase labels occurrences of named entity in the source. Names of the persons, organization, location, and temporal expressions are labeled.

This task has received considerable attention of researchers in last 20 years, and for many domains the resulting models are acceptable. However for historical sources with many entity, names not available in training sets results of the state of the art models are substandard.

Relation extraction

Finally, when entities are recognized in the sources, the last phase is to recognize relationships among them. In the case of the serial historical sources that would often be extraction of family relationships (family reconstruction).

Relation extraction is other hot area of NLP, where excellent results are achieved in the specific domains and on analysis of the modern texts. In general, those systems do not perform so well on historical sources.

First „easy" problem

Parsing proper names

Proper names contain information

Proper names contains information that are tacitly used by human researcher in historiography.

Proper names are hard to parse

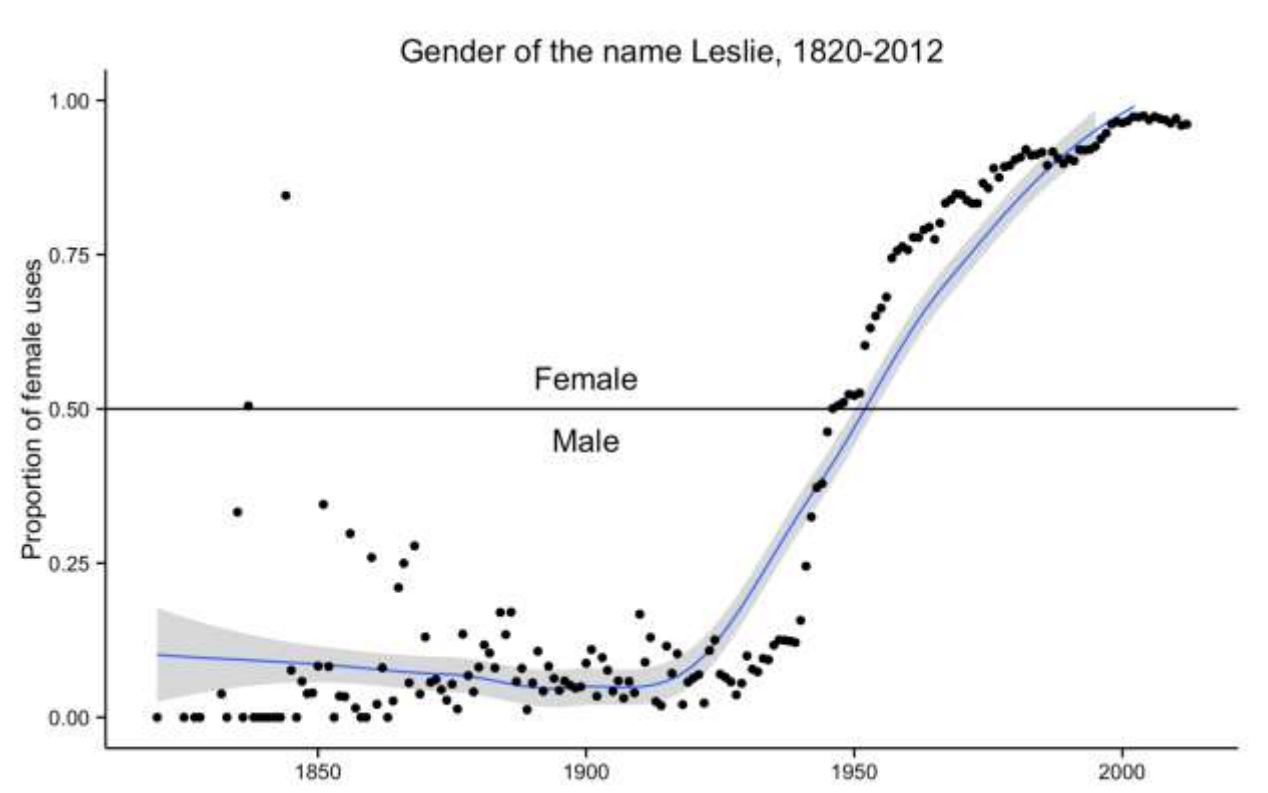
Named entity recognition systems standardly mark only the beginning and the end of the name. Even when more contemporary data sources are involved, the ambiguity, multitude and various combinations of first name/last name/titles that are in use can make this task quite difficult to model.

Understanding proper names helps advanced analysis

Understanding parts of proper names helps more advanced analysis as entity resolution, relation extraction and so o

First „easy" problem

Parsing proper names



(from: Blevins, Mullen: Jane, John … Leslie?, DHQ 9/2015)

Modern examples

(from w3.org - Personal names around the world)

Björk Guðmundsdóttir
Isa bin Osman
毛泽东 (Mao Ze Dong)
María-Jose Carreño Quiñones
José Eduardo Santos Tavares Melo Silva
Борис Николаевич Ельцин
…

Historical examples

Kralj Tomislav (King Tomislav)
Petrasch Marie r. pl. Gemperty od Wiedanthala
Schmidt pl. Silberburg Carl
Hranilović Ferdinand pl. Od Cvietašin
…

Connected problems

Sequence labelling

Input:
Kulmer barunica Josefina r. grofica Oršić

Output:
Kulmer/N.LN-B barunica/N.TITLE-B Josefina/N.FN.F-B
r./N.LN.MAIDEN-B grofica/N.LN.MAIDEN-I Oršić/N.LN.MAIDEN-I

Sequence labelling

Pattern recognition task that labels each member of a sequence with the one of the categorical label.

Part-of-speech tagging (POS)

Sequence labelling task where labels are grammatical parts of speech.

NER

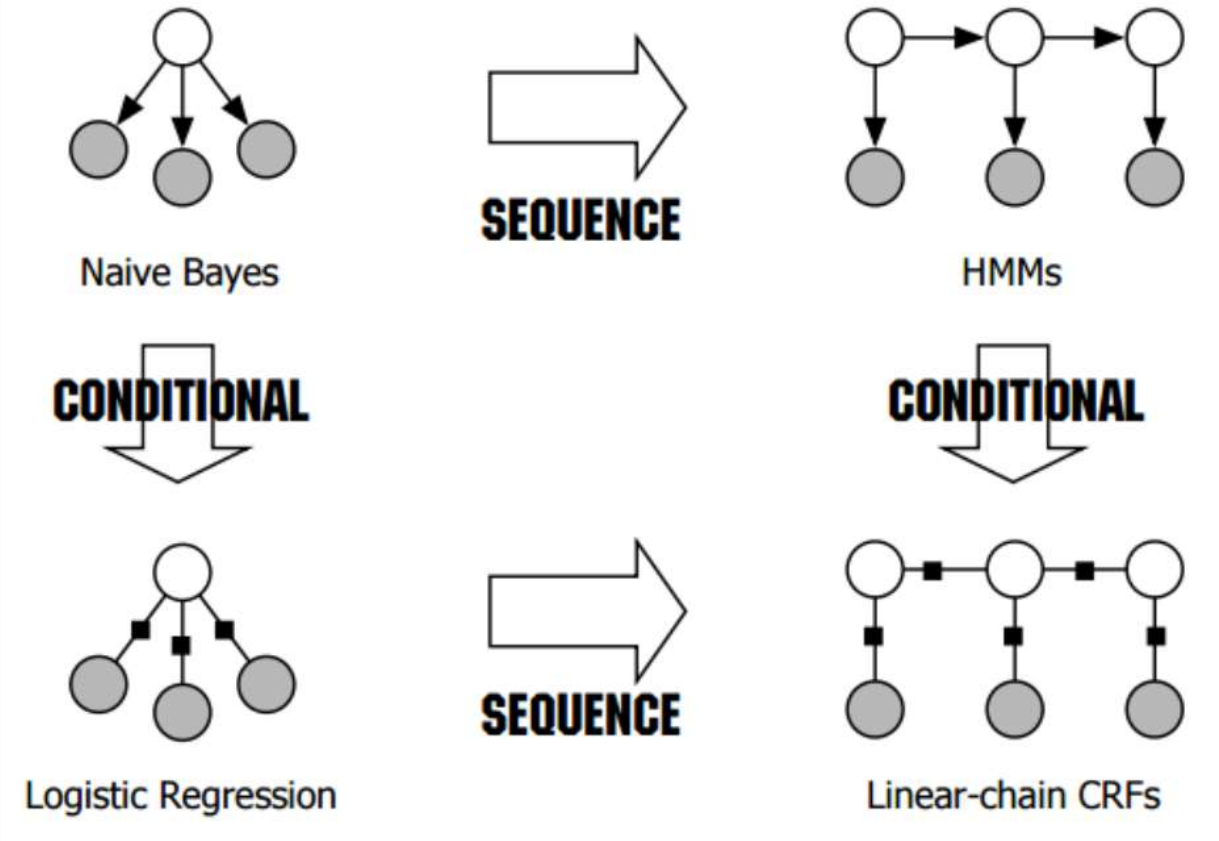
Subtask of information extraction – location and classification of chunks of a text such as the names of persons, locations, organizations, expressions of times and similar

Shallow parsing

Task between full parsing and POS, useful for fast marking of sentence structures such as noun phrases.

State of the art

Conditional random fields



(image source: Sutton, Charles at all, An Intro. To CRF for Rel. Learning)

Rule-based sequence labelling

Brill's Tagger is transformation-based sequence labelling algorithm that use set of predefined rules. **It is one of the first POS model and still widely used.**

Probabilistic graphical models

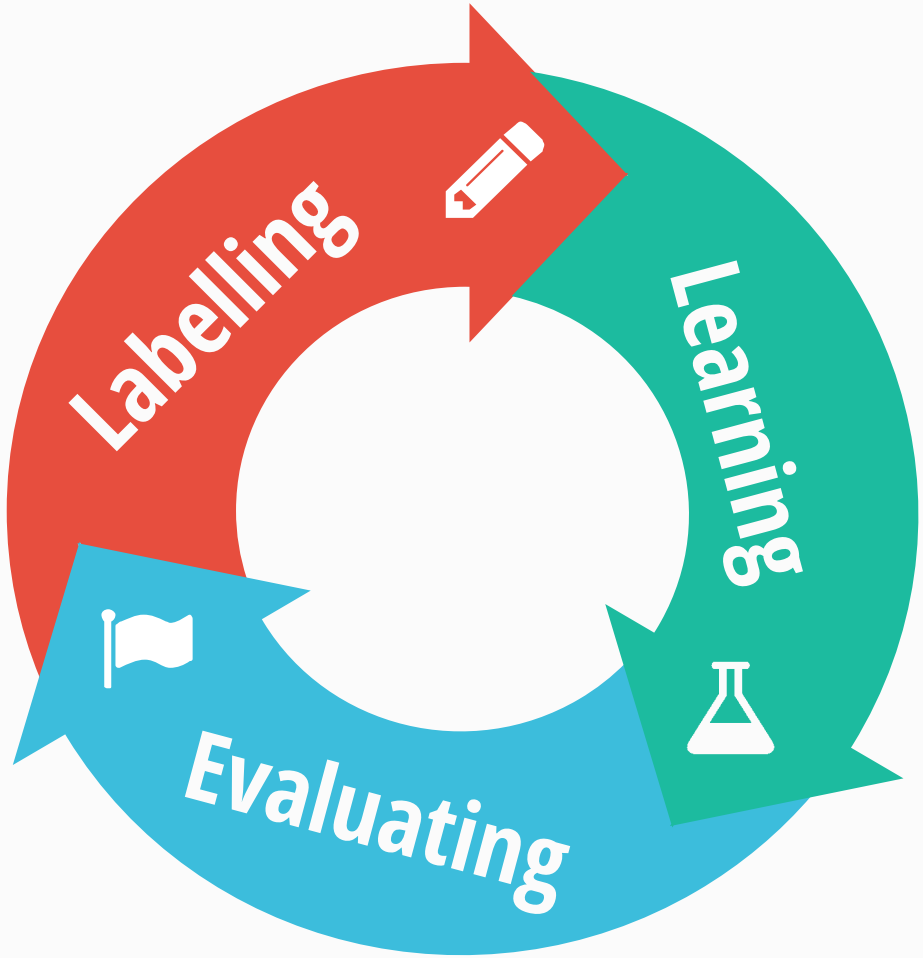
are models for which a graph expresses the conditional probabilistic dependence structure. Most widely used in sequence labelling are Hidden Markov Models and Conditional Random Fields.

Conditional Random Fields (CRF)

CRF are type of probabilistic graphical models, more specifically partial directed Markov Networks. In NLP, the most commonly used type of CRF is a linear-chain CRF.

Machine learning process

For using Conditional Random Fields in parsing



Labelling

Training set is labeled with categories from predefined tag-set.

Learning

One of the learning algorithms is used to create probabilistic model. Cross-validation set is used to adjust parameters of learning algorithm.

Evaluation

Test set is used to evaluate accuracy of the learned model. Accuracy can be measured on token and/or item level.

Labeling - Tagset

(Zagreb 19th cent. Census)

N.FN.(M|F): male/female first name; e.g., *Gustav / Josephine*;

N.LN: last name, e.g., *Philippovich*;

N.LN.PREF: last name, e.g., *de*, *von*;

N.TITLE: person title, e.g., *pl.* (noble), *dr.*;

N.QUAL: surname qualification, e.g., *ml.* (junior);

N.SALUT: person salutation, e.g., *herr* (mister);

GEO: geographic/location term, e.g., *Zagreb*, *Ilica*;

OTHER, terms not in the above list, like notes, comments, etc…

Labelling text

Machine learning process

```
Klaffurich    N.LN-B
pl      N.TITLE-B
.       N.TITLE-I
Eduard N.FN.M-B

Ljubić N.LN-B
Gerga  N.FN.F-B

Mitch  N.FN.M-B
Marcus N.LN-B

Cea    N.LN-B
,       PUNKT
Albert N.FN.M-B

Gemma  N.FN.F-B
Le      N.LN.PREF-B
Marquer       N.LN-B
```

Labelling

Data-set of items (proper names) is tokenized and manually labelled with categories from dataset.

\+

Can be done by persons that are not experts in NLP or computing (students).

\-

Boring and error-prone. In the case of historical text, can be really difficult. Often large training set is needed for satisfactory results.

Learning (training)

Machine learning process

```
def tok2features(sent, i):
    token = sent[i][0]
    lexicon_tag = sent[i][1]
    features = [ 'token.lower=' + token.lower(),
                 'type=' + dict(token),
                 'token[-3:]=' + token[-3:],
                 'token[-2:]=' + token[-2:],
                 'token.isupper=%s' % token.isupper(),
                 'token.istitle=%s' % token.istitle(),
                 'token.isdigit=%s' % token.isdigit(),
                 'lexicon_tag=' + lexicon_tag, 'lexicon_tag[:2]='
                 + lexicon_tag[:2], ]
    if i > 0:
        token1 = sent[i-1][0]
```

Feature extraction

Set of token features is selected for training CRF. In our case the features were:

- Categories (tags) from lexicon with frequency (probability) attached
- Token (normalized)
- Trigrams of token
- Packed representation of case of the token
- Features for token at -1, +1 position

Training

We have used CRFSuite to train the model.

Hyperparameters were optimized using grid search.

Evaluating

Machine learning process

		Predicted					
True		N.FN.M	N.FN.F	N.LN	N.TITLE	…	
	N.FN.M	TP	FN	FN	FN		Recall
	N.FN.F	FP	TP	FN	FN		Recall
	N.LN	FP	FP	TP	FN		Recall
	N.TITLE	FP	FN	FN	TP		Recall
	…						
		Precision	Precision	Precision	Precision		F1-score

Evaluation metrics on token level

Can be evaluated as multiclass classification on token level and cclassification accuracy / f1-score can be used to evaluate the predictions.

Evaluation metrics on item level

Can be evaluated as binary classification on item level.

State of the art approach

Drawbacks

- Crf-s must be trained on a new training set whenever a historical source is systematically different from a previously built model.
- Models are next to impossible to be ad-hoc modified to explore observed regularities in a new domain or historical source.
- In semantically opaque models, there is no easy understandable answer to a “why” question.

Advantages

- The CRF are widely used, and it is easy to use implementations of CRF models.
- Outperforms other models (including HMM,, in many application domains.
- Models can be developed from dataset labelled by persons lacking linguistic and/or computer science skills.

Rule based approach

Machine learning process

```
Brill tagger rules:
(1)VBN VBD PREV-WORD-IS-CAP YES
(2)VBD VBN PREV-1-OR-2-OR-3-TAG HVD
(3)VB NN PREV-1-OR-2-TAG AT
(4)…

MLN rules:
(1)Token(+t,i,c) => Tag(i,+f,c)
(2)Tag(i,+f,c) <=> Tag(i+1,+f,c)
(3)f != f’  => (!Tag(i,+f,c)
   v !Tag(i,+f’,c))
```

Brill tagger

1. Assign most frequent tag
2. Transformation based: Rules of the form

tag1 → *tag2* IF *Condition*

are used to transform tags

Markov Logic Networks

Markov Logic Networks (MLN) generalize first-order logic (FOL).

Weights (probabilities) are attached to FOL statements.

Rule based approach

Drawbacks

- If rules are hand-coded it has to be done by researchers, trained and experienced in both domain specific knowledge & a rule-based system.
- Learning algorithms are inferior to the ones used to train statistical models.
- Complex interaction of rules can make difficult understanding and ad-hoc modification of the rules.

Advantages

- Possibility of coding general & domain specific constraint & rules .
- Learning can be performed on the top of hand-coded rules.
- Resulting rules are relatively semantically transparent and can be ad-hoc modified and improved.

Default model semantics

Rule based approach

```
tag("david",n.fn.m) :- not tag("david",n.ln).
tag("david",n.ln) :- not tag("david",n.fn.m).


swim :- not sharks.
swim :- ~ sharks.
```

Answer Set Programming

(ASP) is a form of logic programming based on the stable model (answer set) semantics.

Stable model semantics

An approach to define semantics of negation in logic programming, declarative semantics for logic programs with negation as failure.

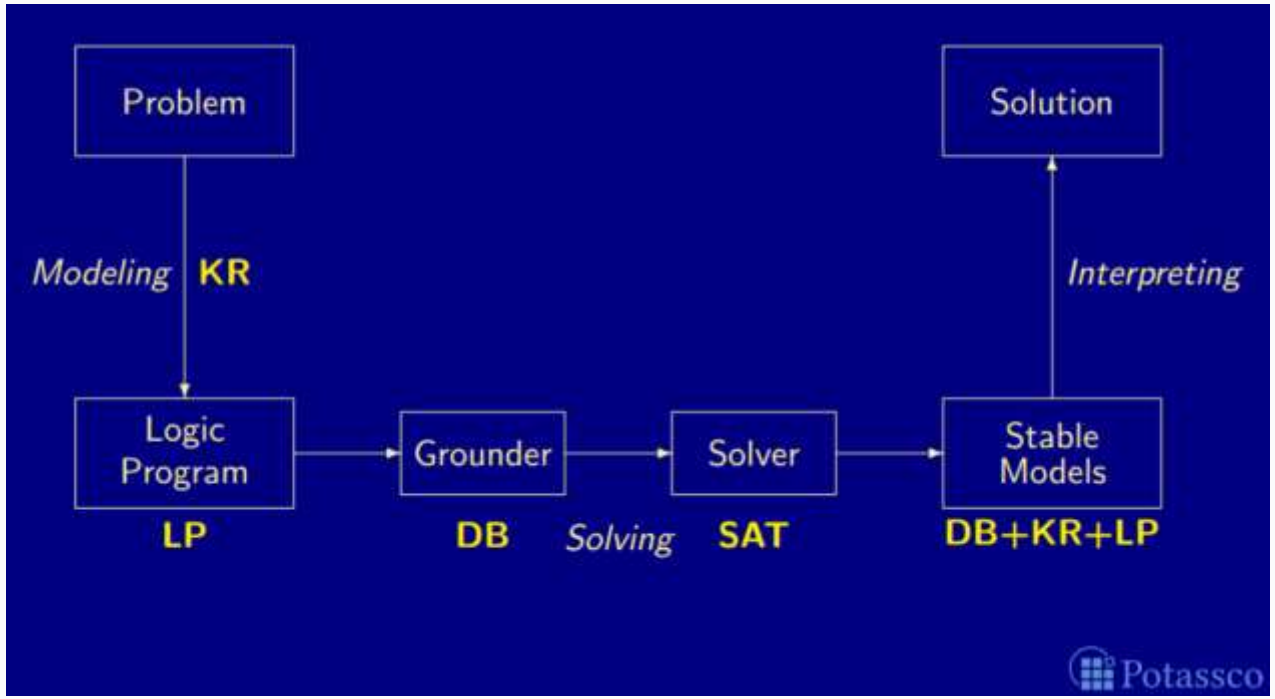
Possibility to model two types of negation: negation as failure and strong -> useful in modelling incomplete knowledge.

Implementation

Potassco - the Potsdam Answer Set Solving Collection

Answer Set Programming

Rule based approach



(image source: T. Schaub: Answer set solving in practice)

Modeling

ASP enables declarative modelling of the problem (sequence labelling).

Grounder

LP rules are grounded - replace with rules without variables.

Solver

Optimised and efficient SAT solvers are used to generate results.

Lexicon

Rule based approach

TOKEN	TAG	LANG	SOURCE	SOURCE_ID	CNT	CNT_CORP	EXISTS_SEPARATE
Szklarek	N.LN	en	viaf	(null)	1	17151761	(null)
Susegg	N.LN	(null)	viaf	(null)	1	17151761	(null)
Žepek	N.LN	hr	viaf	(null)	1	17151761	(null)
Bleyberg	N.LN	de	viaf	(null)	2	17151761	(null)
Smullen	N.LN	ga	viaf	(null)	1	17151761	(null)
Warakomska	N.LN	pl	viaf	(null)	5	17151761	(null)
Nekritin	N.LN	en	viaf	(null)	1	17151761	(null)

Name lexicon

Large lexicon of name parts is built from public available sources to enhance name parts labeling.

Features of the lexicon

Number of records: **26.8 millions**
Number of distinct tokens: **12.3 millions**
Number of tags: **10**
Number of languages: **152**
Number of sources: **16**

Accuracy of lexicon

On modern text samples: 62-75%
On the historical sample: 53%

Representing rules

Rule based approach

```
tag(I,P,[tag],[weight], [level])

tag(I,P,n_title_b,70,1) :-
       lexc(I,P,n_title_b,_,_),
       tokenform(I,P,"LlLlLl"),
       tokenform(I,P-1,"LuLlLl"),
       lexc2(I,P-1,n_fn,_,1),
       lexc2(I,P+1,n_ln,_,1),
       not specExists(I,P,1).
```

Predicates (facts)

Text is encoded with predicate:

```
Token(record_number, token_position, string_of token).
```

Features

Features of the token is encoded as follows:

```
lexc(record_no,token_pos,tag,probability,prob_rank).
tokenform(record_no,token_pos,packed_case_repress).
begin(record_no, token_pos).
end(record_no,token_pos).
```

Lexc2 is defined as shorthand for lexc.

Non-monotonicity

Special predicate “specExists” is defined to use non-monotonic inference, and enables modelling rules from general to more specific.

Learning rules

Rule based approach

```
Generate all features of examples in the training set
Generalize features [replace constants with variables, relativize positions]
Select top-n features (eliminate all with low chi-square in the training set)
for lev in 1 to maxLevel
        predicted = tag training set with rules up to level lev-1
        for tag in tag-set
                for x in power set of features up to length maxCardinality
                        gain = count false negative matching x in predicted
                        loss = count true negative matching x in predicted
                        if gain>loss & gain-loss>minGen add x to rules candidates
                for y in rules candidates sorted by gain-loss
                        if rules does not overlap with rules add y to rules
```

Learning

A few of initial rules were hand-coded, and the rest of rules was learned from 6.350 labelled modern-text items (name expressions) . The system generated 218 rules in 4 levels of generality

Evaluation

On the test subset of modern text dataset (20%) , average accuracy rate on token level was 0.947 .

Application to Zagreb 1857 census

Rule based & statistical approach

Aichelburg Freiher Ernst
Aichelburg Theres
Aichelburg Hedvig
Aichelburg Hermine
Aichelburg Richard
Allovich pl. Julie
Bertele Marie pl. Grenadenberg
Blagaj pl. Berta
Blagaj Alois
Blagaj Christina
Blagaj Maria
Bunijevac pl. Jos
Kuković v. Slavomir
Cinkhemer pl. Franz
Conrad v. Eibersfeld Sigmund
Conrad Wilhelmine
Conrad Ema
Conrad Fritz
Conrad Helene
Conrad Heinrich
Conrad Hugo
Conrad Maria
Cvetković v. Karl
Czernkovich Joh. Nep. Vn Dolje
Čačković Franjo pl. Verhovinski
Čačković Katarina rođ. Jelačić

Census in the year 1857

Part of the census data was available in transcribed form – total of 1775 name expression.

Application of CRF

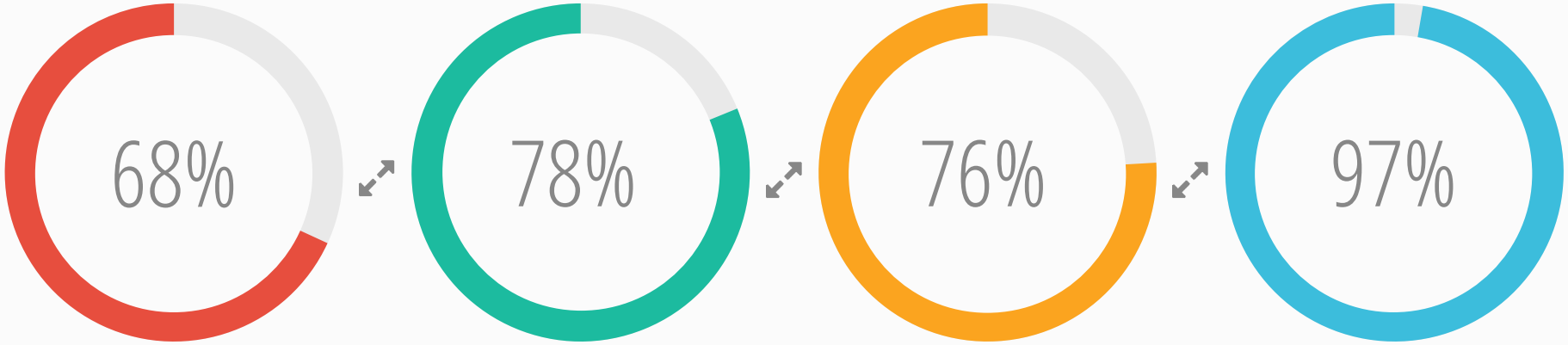
Accuracy rate of CRF trained on the modern text was 68%.

Application of default semantics rules

Accuracy rate of rules based system trained on the modern text was 78%.

Comparative results inferring name form

Statistical & rule based approach



CRF

Initial

Labelled using model trained on the modern data.

Rules

Initial

Labelled using model trained on the modern data.

CRF

Improved

The CRF model was improved by labelling 100 addition records from the source and adding these to the dataset.

Rules

Improved

The SM Rules model was improved by hand-writing four additional rules that were obvious from the errors in the first model.

Gender & nobility status classifiers

Lessons learned

Strengths

First results indicated that the rule-based approach, which was based on stable model semantics, is more suitable for inferring standard name form from historical text than the more widespread statistical approach.

Weaknesses

To confirm this result, the experiment should be replicated by using additional historical sources and statistical models.

Potentials

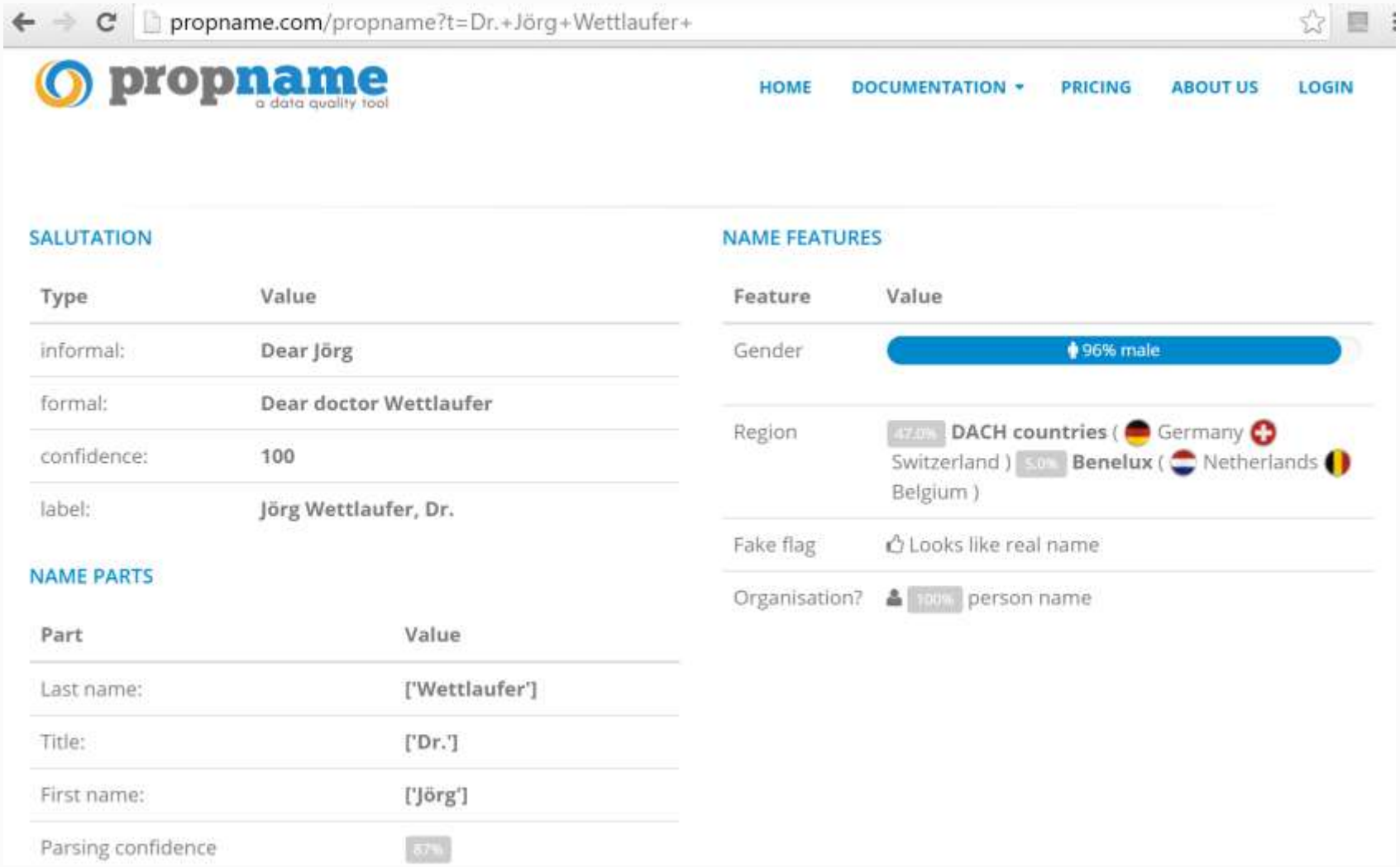
Model ensemble that includes both a rule-based method and the CRF model is another interesting development that is worth future research.

Further achievements

The development of a more complex system that includes joint inference from the scan of a source to a historical demography web ontology is a worthwhile longer-term goal. This research represents a small step toward the development of such a system.

Further achievements – propname.com

Applications to modern names



Further achievements

Applications of name to digital humanities research

Proper name category	#Entities	#Tokens	Sources
Person full name	160.4M	n/a	Wikidata,VIAF Whitepages web scrape
First names	-||-		-||-
Last names	-||-		-||-
Geographic name	1.1M	n/a	Geonames, Wikidata
Organisation name	13.6M	48.4M	Wikidata, Freebase, Web scrape

Countries classification based on proper name similarities

Classification of countries & languages is based on generalized Jaccard similarity among proper names.

Sim. to DE	FN	LN	COMP
AT	0.18	0.27	0.24
CH	0.19	0.32	0.18
NL	0.16	0.33	0.19
ZA	0.13	0.24	0.22
BE	0.12	0.28	0.16

Further achievements

Applications to digital humanities research

Gender Differences in Authorship of Book Topics

Topics	Male	Female	M_REL	F_REL	Difference
Programming	1,704,918	2,458	1.11%	0.00%	43288%
Dr. Seuss	71	59,959	0.00%	0.04%	26008%
Exports & Imports	513,187	1,850	0.33%	0.00%	15341%
Civil Service	190,914	153	0.12%	0.00%	11303%
Studying & Workbooks	600,677	10,898	0.39%	0.01%	4934%
Machinery	125,336	1,268	0.08%	0.00%	4513%
Western & Frontier	630	34,496	0.00%	0.02%	4294%
Greek & Roman	177,255	3,712	0.12%	0.00%	3434%
Romance	9,663	293,762	0.01%	0.19%	2920%
Test Preparation	90,650	1,616	0.06%	0.00%	2908%
Continental European	173,110	4,958	0.11%	0.00%	2714%
Repair & Maintenance	62,088	965	0.04%	0.00%	2502%
Chess	58,451	823	0.04%	0.00%	2495%
Greek & Roman	102,447	2,715	0.07%	0.00%	2445%
Essays	68,231	1,297	0.04%	0.00%	2432%
Contemporary	3,619	87,230	0.00%	0.06%	2256%
Guitar	100,582	3,034	0.07%	0.00%	2235%
Time Management	10,805	246,317	0.01%	0.16%	2193%
Automotive	58,231	1,151	0.04%	0.00%	2187%
Mystery & Suspense	2,526	57,421	0.00%	0.04%	2091%
Regency	14,617	307,864	0.01%	0.20%	2034%
Image Comics	40,311	457	0.03%	0.00%	2030%
Aviation	65,426	1,727	0.04%	0.00%	2029%
British & Irish	666,817	32,416	0.43%	0.02%	2010%
Historical	4,485	94,197	0.00%	0.06%	1982%

Discussion Time!

- [x]

Thanks for Watching

Darko Vitek (dvitek@hrstud.hr); Davor Lauc (dlauc@ffzg.hr)