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
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

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
Historical 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.
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.

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 satisfactorily on standard modern text; but no so well on historical sources.

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.

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 contain information that are tacitly used by human researchers 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 on.
First „easy” problem
Parsing proper names

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 Wiedanthal
Schmidt pl. Silberburg Carl
Hranilović Ferdinand pl. Od Cvietašin

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

Rule-based sequence labelling
Brill’s tagger is a transformation-based sequence labelling algorithm that uses a set of predefined rules. It is one of the first POS models 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.

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

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>N.FN. (M</td>
<td>F)</td>
<td>male/female first name; e.g., Gustav / Josephine;</td>
</tr>
<tr>
<td>N.LN</td>
<td>last name, e.g., Philippovich;</td>
<td></td>
</tr>
<tr>
<td>N.LN.PREF</td>
<td>last name, e.g., de, von;</td>
<td></td>
</tr>
<tr>
<td>N.TITLE</td>
<td>person title, e.g., pl. (noble), dr.;</td>
<td></td>
</tr>
<tr>
<td>N.QUAL</td>
<td>surname qualification, e.g., ml. (junior);</td>
<td></td>
</tr>
<tr>
<td>N.SALUT</td>
<td>person salutation, e.g., herr (mister);</td>
<td></td>
</tr>
<tr>
<td>GEO</td>
<td>geographic/location term, e.g., Zagreb, Ilica;</td>
<td></td>
</tr>
<tr>
<td>OTHER</td>
<td>terms not in the above list, like notes, comments, etc...</td>
<td></td>
</tr>
</tbody>
</table>
Labelling text

Machine learning process

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

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.

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]
Evaluating Machine learning process

<table>
<thead>
<tr>
<th>True</th>
<th>Predicted</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N.FN.M</td>
<td>N.FN.F</td>
<td>N.LN</td>
<td>N.TITLE</td>
<td>...</td>
</tr>
<tr>
<td>N.FN.M</td>
<td>TP</td>
<td>FN</td>
<td>FN</td>
<td>FN</td>
<td>Recall</td>
</tr>
<tr>
<td>N.FN.F</td>
<td>FP</td>
<td>TP</td>
<td>FN</td>
<td>FN</td>
<td>Recall</td>
</tr>
<tr>
<td>N.LN</td>
<td>FP</td>
<td>FP</td>
<td>TP</td>
<td>FN</td>
<td>Recall</td>
</tr>
<tr>
<td>N.TITLE</td>
<td>FP</td>
<td>FN</td>
<td>FN</td>
<td>TP</td>
<td>Recall</td>
</tr>
<tr>
<td>...</td>
<td>Precision</td>
<td>Precision</td>
<td>Precision</td>
<td>Precision</td>
<td>F1-score</td>
</tr>
</tbody>
</table>

Evaluation metrics on token level
Can be evaluated as multiclass classification on token level and classification 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

**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.

**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.
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

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

```prolog
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
Rule based approach

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.

(image source: T. Schaub: Answer set solving in practice)
Lexicon
Rule based approach

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

<table>
<thead>
<tr>
<th>tag(I,P,[tag],[weight],[level])</th>
</tr>
</thead>
<tbody>
<tr>
<td>tag(I,P,n_title_b,70,1) :-</td>
</tr>
<tr>
<td>lexc(I,P,n_title_b,,_),</td>
</tr>
<tr>
<td>tokenform(I,P,&quot;LlLlLl&quot;),</td>
</tr>
<tr>
<td>tokenform(I,P-1,&quot;LuLlLl&quot;),</td>
</tr>
<tr>
<td>lexc2(I,P-1,n_fn,,1),</td>
</tr>
<tr>
<td>lexc2(I,P+1,n_ln,,1),</td>
</tr>
<tr>
<td>not specExists(I,P,1).</td>
</tr>
</tbody>
</table>

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

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.

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
Application to Zagreb 1857 census
Rule based & statistical approach

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**: 68%
  - Labelled using model trained on the modern data.

- **Rules Initial**: 78%
  - Labelled using model trained on the modern data.

- **CRF Improved**: 76%
  - The CRF model was improved by labelling 100 additional records from the source and adding these to the dataset.

- **Rules Improved**: 97%
  - The SM Rules model was improved by hand-writing four additional rules that were obvious from the errors in the first model.

Labelled using model trained on the modern data.
Gender & nobility status classifiers

Gender
99%
Gender classifier accuracy, f1-score 98%

Nobility
96%
Nobility status classifier accuracy
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 – proppname.com

Applications to modern names

<table>
<thead>
<tr>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SALUTATION</td>
<td></td>
</tr>
<tr>
<td>informal</td>
<td>Dear Jörg</td>
</tr>
<tr>
<td>formal</td>
<td>Dear doctor Wetlauffer</td>
</tr>
<tr>
<td>confidence</td>
<td>100</td>
</tr>
<tr>
<td>label</td>
<td>Jörg Wetlauser, Dr.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NAME FEATURES</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature</td>
<td>Value</td>
</tr>
<tr>
<td>Gender</td>
<td>95% male</td>
</tr>
<tr>
<td>Region</td>
<td>DACH countries (Germany + Switzerland) + Benelux (Netherlands + Belgium)</td>
</tr>
<tr>
<td>Fake flag</td>
<td>Looks like real name</td>
</tr>
<tr>
<td>Organisation?</td>
<td>100% person name</td>
</tr>
</tbody>
</table>

SALutation

NAME FEATURES
Further achievements
Applications of name to digital humanities research

Countries classification based on proper name similarities
Classification of countries & languages is based on generalized Jaccard similarity among proper names.

<table>
<thead>
<tr>
<th>Sim. to DE</th>
<th>FN</th>
<th>LN</th>
<th>COMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT</td>
<td>0.18</td>
<td>0.27</td>
<td>0.24</td>
</tr>
<tr>
<td>CH</td>
<td>0.19</td>
<td>0.32</td>
<td>0.18</td>
</tr>
<tr>
<td>NL</td>
<td>0.16</td>
<td>0.33</td>
<td>0.19</td>
</tr>
<tr>
<td>ZA</td>
<td>0.13</td>
<td>0.24</td>
<td>0.22</td>
</tr>
<tr>
<td>BE</td>
<td>0.12</td>
<td>0.28</td>
<td>0.16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Proper name category</th>
<th>#Entities</th>
<th>#Tokens</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person full name</td>
<td>160.4M</td>
<td>n/a</td>
<td>Wikidata, VIAF Whitepages web scrape</td>
</tr>
<tr>
<td>First names</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Last names</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Geographic name</td>
<td>1.1M</td>
<td>n/a</td>
<td>Geonames, Wikidata</td>
</tr>
<tr>
<td>Organisation name</td>
<td>13.6M</td>
<td>48.4M</td>
<td>Wikidata, Freebase, Web scrape</td>
</tr>
</tbody>
</table>
### Further achievements

Applications to digital humanities research

#### Gender Differences in Authorship of Book Topics

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