Using Web Scraped Data to Enhance the Quality of the Statistical Business Register

Manveer Mangat, Heidi Kühnemann,
Arnout van Delden, Johannes Gussenbauer

Trusted Smart Statistics – Web Intelligence Network
Grant Agreement: 101035829

Web Intelligence Network

Funded by the European Union

Outline

- Webscraping
- URL-Finding
- Link web scraped 3rd party data to the SBR
- NACE Code Classification
  - Introduction & Case Study Statistics Netherlands
  - Case Study Statistics Austria
  - Lessons Learned

Webscraping

Heidi Kühnemann
Statistics Hesse

Web Intelligence Network

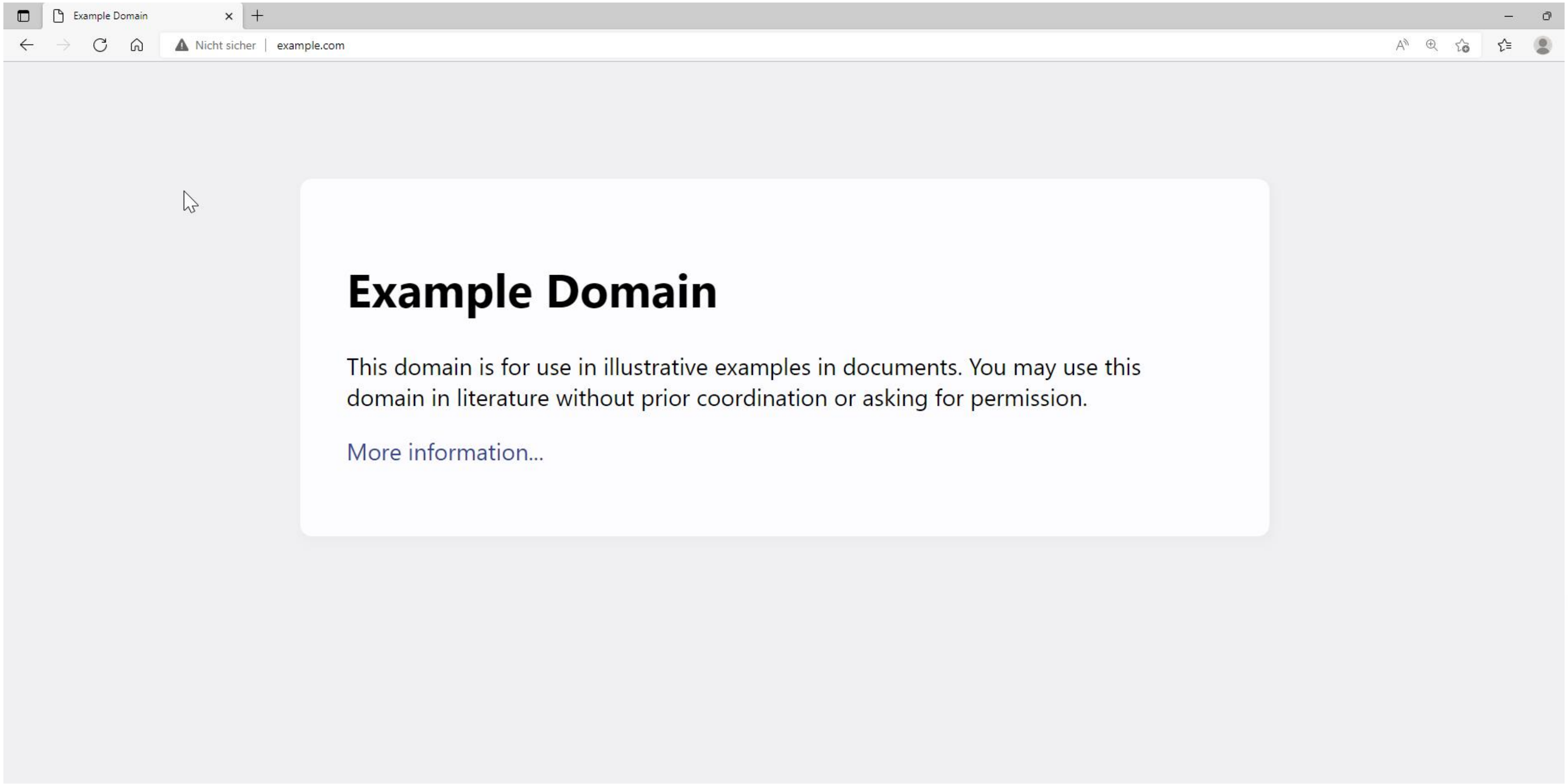
Funded by the European Union

Web scraping

- Definition: Automated gathering of data from the world wide web
- Examples for web data sources
  - Search engines
  - Online Shops
  - Hotel booking platforms
  - Enterprise websites
  - Social media
  - News websites
  - Personal blogs
  - Wikipedia

Official Statistics mostly focusses on these

What we see


Example Domain
Nicht sicher
example.com
Example Domain
This domain is for use in illustrative examples in documents. You may use this domain in literature without prior coordination or asking for permission.
More information...

Example Domain | view-source:example.com

Nicht sicher | view-source:example.com

Zeilenumbruch

```html
<!doctype html>
<html>
<head>
    <title>Example Domain</title>

    <meta charset="utf-8" />
    <meta http-equiv="Content-type" content="text/html; charset=utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <style type="text/css">
    body {
        background-color: #f0f0f2;
        margin: 0;
        padding: 0;
        font-family: -apple-system, system-ui, BlinkMacSystemFont, "Segoe UI", "Open Sans", "Helvetica Neue", Helvetica, Arial, sans-serif;

    }
    div {
        width: 600px;
        margin: 5em auto;
        padding: 2em;
        background-color: #fdfdff;
        border-radius: 0.5em;
        box-shadow: 2px 3px 7px 2px rgba(0,0,0,0.02);
    }
    a:link, a:visited {
        color: #38488f;
        text-decoration: none;
    }
    @media (max-width: 700px) {
        div {
            margin: 0 auto;
            width: auto;
        }
    }
    </style>
</head>

<body>
<div>
    <h1>Example Domain</h1>
    <p>This domain is for use in illustrative examples in documents. You may use this
    domain in literature without prior coordination or asking for permission.</p>
    <p><a href="https://www.iana.org/domains/example">More information...</a></p>
</div>
</body>
</html>
```

vs. What we scrape

Specifc vs. Generic web scraping

Specific web scraping	Generic web scraping
Website structure is known	Website structure is **not** known
Extraction of specific elements in HTML code (eg. with XPATH, css selectors)	Text mining, regular expressions, etc. to extract information
Extracted data usually contains the information of interest	Extracted data by itself is often not very meaningful, but is input for further models

This is what we focus on today

URL-Finding

Heidi Kühnemann
Statistics Hesse

Web Intelligence Network

Funded by the European Union

Enterprise URLs: Why and How?

Why?

- Freely available enterprise information on various topics
- Potential to reduce response burden in some areas
- Potential to update statistical business register (SBR) with additional data source

How?

- Obtain data from registers and surveys (not always possible!)
- Data purchases → Topic 3
- **Automated procedure to search for URLs**

URL finding overview

Search enterprises in search engine

Search term, eg. name + municipality

URL patterns to include/exclude in API settings

Exclusion of some file types in search term

Scrape URLs

Scraped are eg.:

Google URL

Landing page of domain

Imprint/contact page

Feature Extraction

Features:

- Name
- Address
- Tax/business register IDs
- Search position

Some text processing steps

ML / Deterministic rules

Needs training data!

Manually classified or obtained through other means

Search engines

Criteria to consider when selecting a search engine:

- Can a SE identify the correct URLs?
- Limits in the number of requests
- Costs of requests

% domains matched	GOOGLE	GOOGLE API	BING	YAHOO	DUCK
Italian sample	74.8	66.7	64.7	63.6	57.6
Hessian sample	89	87	62	59	NA

Comparison of SE results for ca. 100 Italian and Hessian enterprises

API or Search Engine Scraping?

API:

✓ Many configuration options

✓ High frequency of requests possible

× Only small number of requests are free

Search Engine Scraping:

✓ Requests are free

✓Obtain results like a human being

× Potential violation of terms of use

× Scrapers might get blocked

Scraping

- By far the most cumbersome step: scrape all result URLs
- Each search produces ca. 10-30 URLs to be scraped (result URLs, contact pages, imprint, landing,…)
- URLs are very diverse: different technologies, sometimes large contents
- Information is sometimes hidden in Javascript → Javascript rendering software is advisable (automated browser)
- Headless browsers: Selenium or Splash are in use within ESS
- But: Javascript rendering increases the amount of downloaded data and bandwidth usage
- Massive scraping needs special infrastructure

Feature Extraction

- Preprocessing steps, eg.
  - remove css styles and javascript code
  - remove duplicate whitespaces
  - lowercasing words and letters
- Compare enterprise data from SBR with scraped data, eg.
  - Name is on website
  - VAT ID is on websites,
  - …
- Features are created with exact string matching or regular expressions
- String similarity for comparison of short texts with enterprise data (eg. name and HTML title)

Machine Learning / Deterministic Rules

- When do we accept a URL as correct?
- Deterministic rules:
  - eg. VAT ID on website → website correct
  - Easy to build and interprete
  - What if enterprise data is missing in the SBR or on the website?
  - What if other website mentions data of different enterprises?
  - Validation data necessary to measure classification performance
- Machine Learning:
  - Training & validation data necessary
  - Model decides which features have which weight
  - Reduced interpretability

Funded by the European Union

URL finder software

- Python (Statistics Netherlands): https://github.com/SNStatComp/urlfinding
- Python (Statistics Bulgaria): https://github.com/EnterpriseCharacteristicsESSnetBigData/StarterKit/tree/master/URLsFinder
- Java (Istat): https://github.com/EnterpriseCharacteristicsESSnetBigData/UrlSearcher
- R (Statistics Hesse): Not yet published

Literature / Further reading

WIN Report on URL finding methodology (https://ec.europa.eu/eurostat/cros/system/files/20220131_url_finding_methodology.pdf)

Delden, Arnout van; Windmeijer, Dick; Bosch, Olav ten (2019): Searching for business websites. CBS (Discussion Paper). https://www.cbs.nl/en-gb/background/2020/01/searching-for-business-websites.

Barcaroli, Giulio; Scannapieco, Monica; Summa, Donato (2016): On the Use of Internet as a Data Source for Official Statistics: a Strategy for Identifying Enterprises on the Web. In: Italian Review of Economics, Demography and Statistics 4 (70), S. 25–41. http://www.sieds.it/listing/RePEc/journl/2016LXX_N4_RIEDS_25-41_Scannapieco.pdf.

How to link web scraped 3rd party data to the business register

Arnout van Delden, Nick de Wolf
Statistics Netherlands

Web Intelligence Network

Funded by the European Union

Introduction

- Third parties web scraped business data:
  URL, economic activity, phone number, keywords website text, …
- Aim to link URL to ‘businesses’ in Statistical Business Register
- SBR: legal units (LUs) are building blocks
- Therefore it is practical to link URLs to LUs

- Often LU – website links are 1:1
- Sometimes multiple URLs link to a LU (e.g. different products)
- Sometimes a URL links to multiple LU’s (enterprise group).

Example SN: building a linkage approach

- 3rd party data: Dataprovider (DP)
- What identification keys are in both sources?

Key	Type of key	Reliable?
Hostname (*)(**)	Unique	High
Domain name (**)	Unique	High
LU–ID	Unique	High
Email	Non-unique	Medium
Zip-code	Non-unique	Medium / Low
Phone	Non-unique	Medium / Low

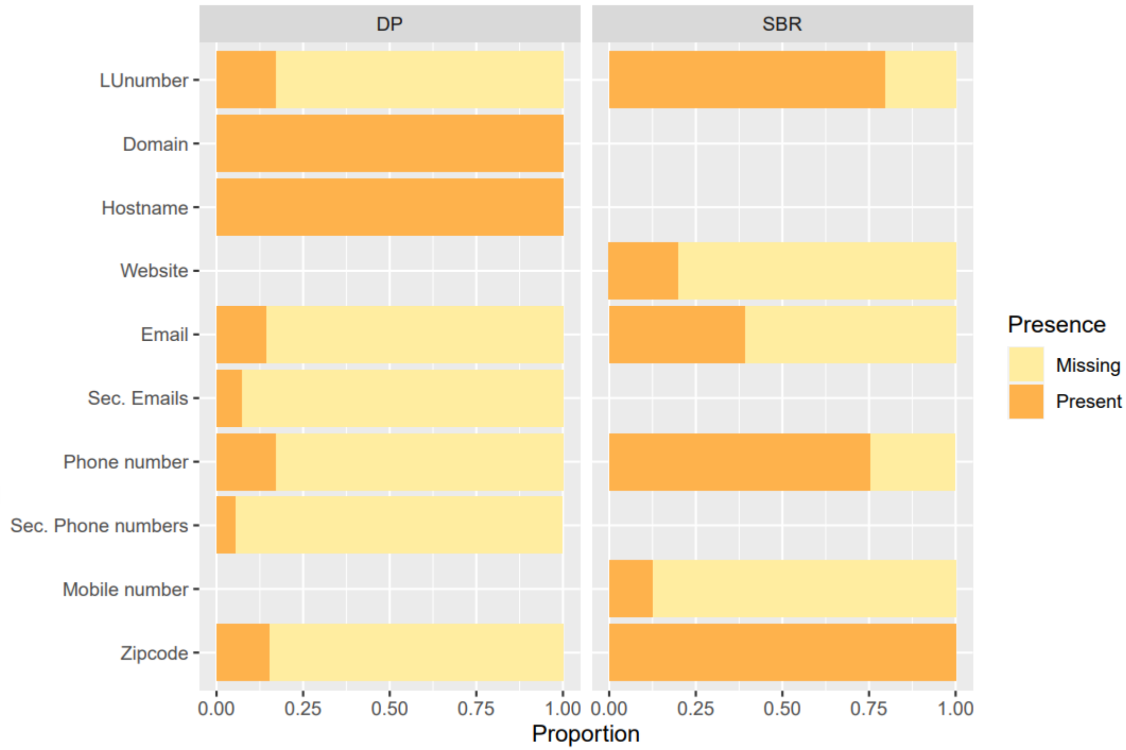
(*) when a LU registers at Chamber of Commerce it may mention the URL. That is sent to Statistics Netherlands.

(**) hostname: dashboards.cbs.nl, domain name: cbs.nl

Analysis: missing information

- Considerable part of the identifying information is missing

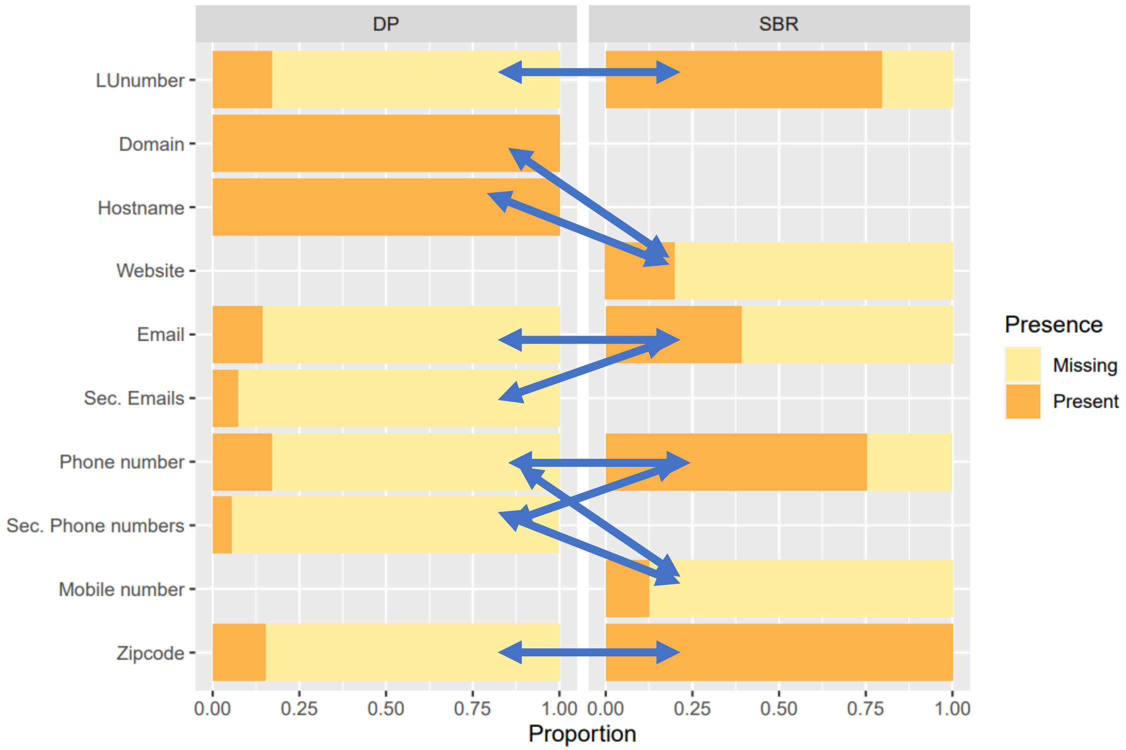
(counted in Oct 2020)



Analysis: missing information

- Considerable part of the identifying information is missing

(counted in Oct 2020)



Linkage approach

Development of linkage protocol:

1. Stepwise linkage procedure with qualitative score function (2016)(*)
2. More generic linkage approach based on agreement of linkage keys with approximate linkage probability based on points (2019)
3. As 2, but now the linkage probability is based more advanced regression model and evaluation of linkage quality (2022-2023)

(*) see Oostrom, L. et al. (2016). Measuring the internet economy in The Netherlands: a big data analysis. CBS Discussion paper 2016-14 (publicly available)

Funded by the European Union

2 Linkage using linkage points (2019)

- Agreement per variable is given certain linkage points
- Points based on 2016 protocol and trial and error
- 20 links checked per “total number of points”-category to estimate linkage probability: 47.5*LN(points) – 234, with min = 0, max = 1.

Total number of points	Linkage probability
100-300	0
400	50
500-900	75
1000	95
1100	97
>=1200	100

DataProvider	Legal unit	Points
Hostname	Website	500
Domain	Website	500
LU-ID	LU-ID	500
Email	Email	200
ZipCode	ZipCode	200
Phonenumber	Phonenumber	100

Web Intelligence Network

Funded by the European Union

3 Linkage probability (2022-2023)

- Sample of potential matches is evaluated (400 per group):
  - linkage probability < 50
  - Linkage probability >=50
  - 3rd party hostnames not linked
  - SBR LUs not linked

- Estimate a (weighted) logistic regression model with probability of a match (yes/no) as a function of agreement (yes/no) per variable[1]
- Results on non-matched records may lead to more variables that are used as linkage keys.

[1] Tuoto (2016). New proposal for linkage error estimation. Statistical journal of IAOS 32 (2016) 413–420

Web Intelligence Network
Funded by the European Union

Results (2020): type of linkage

- One URL may link to multiple legal units:
  - e.g. website of an air plane company that refers to enterprise group
- A legal unit may link to multiple URLs:
  - e.g. different products on different websites
- At 75% linkage probability:

<table>
<tr><th></th><th></th><th># ULRs</th><th></th><th></th></tr>
<tr><td># LU’s</td><td>2+ (n)</td><td>1</td><td>0</td><td>Total</td></tr>
<tr><td>2+ (m)</td><td>4 863</td><td>27 935</td><td rowspan="2">3 957 354</td><td rowspan="2">4 630 836</td></tr>
<tr><td>1</td><td>111 904</td><td>528 780</td></tr>
<tr><td>0</td><td colspan="2">5 057 922</td><td>X</td><td>X</td></tr>
</table>

Funded by the European Union

Introduction on how to apply automatic prediction of NACE codes from web scraped texts

Arnout van Delden, Nick de Wolf
Statistics Netherlands

Web Intelligence Network

Funded by the European Union

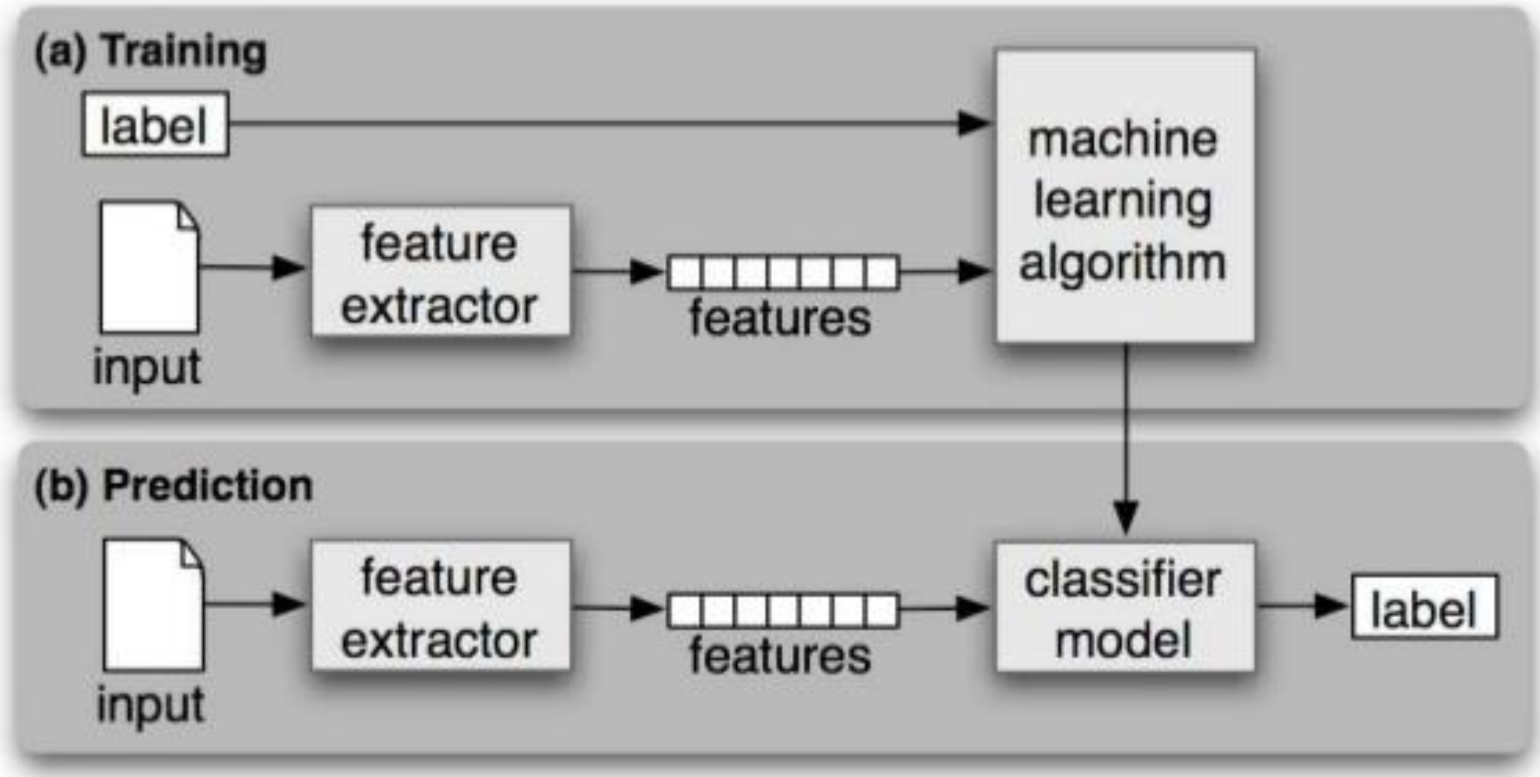
Approach

- There is no standard recipe for NACE prediction using website texts
- Situations per country differ in many ways (purpose, language,)

Therefore:

- Purpose is to inspire and to share lessons learned.
- We like to learn from you when you have tried it yourself
- We will share experiences from Statistics Netherlands and Austria
- We show different steps of the process and choices that we made…

Elements to consider



Input, feature extractor

Input

- URL, headers, Main body, subpages (about us, contact page)

Extract features

- to extract useful text parts, remove HTML (Justext)
- keep only texts of language(s) of interest (Langdetect)
- drop uninformative websites: HTML-errors and texts like ‘this domain is reserved’ or ‘this domain is unavailable’

Funded by the European Union

Actual features

1. Pre-processing: downcasing, stopword removal, stemming or lemmatisation
2. Selection of tokens: Knowledge-based features, use of global feature importance
3. Weighting of tokens: Tf-idf weighting, BM25 weighting
4. Adding context via word-embeddings: (Fasttext, doc2vec)

Machine learning algorithms

1. Many different algorithms are available: classical textmining / neural-net algorithms
2. Hierarchical versus direct prediction the NACE code at the level of interest
3. Hyperparameter tuning very important

Labels, train and test set

1. What NACE level and what which spectrum of codes?

2. How can you obtain a (nearly) error-free data set?
   - Erroneous labels are learned by the ML model, so should be avoided

3. Balancedness of the train set:
   - With unbalanced set more difficult to achieve an accurately trained model

4. Is your test set representative of the targeted population?
   - Ideally, one has inclusion weights with respect to the population

Evaluate model performance

1. What kind of predictions are you interested in?
   - A single label per unit, multiple labels and / or a probability per label
2. Where do you use the predictions for?
   - Support manual editing or automatically predict new labels?

3. Performance per record or per NACE most important?

4. Do not forget the confusion matrix

Case study Stats Netherlands

Aim: Predict main activity of legal units

Data: 35 733 URLs in NACE section R, homepage, `About us', `Contact' or `Terms and Conditions' page, plus up to 10 underlying pages

Knowlegde based features: *concept words* (C-words; car) and *descriptive words* (D-words; station wagon, four-wheel drive, …)

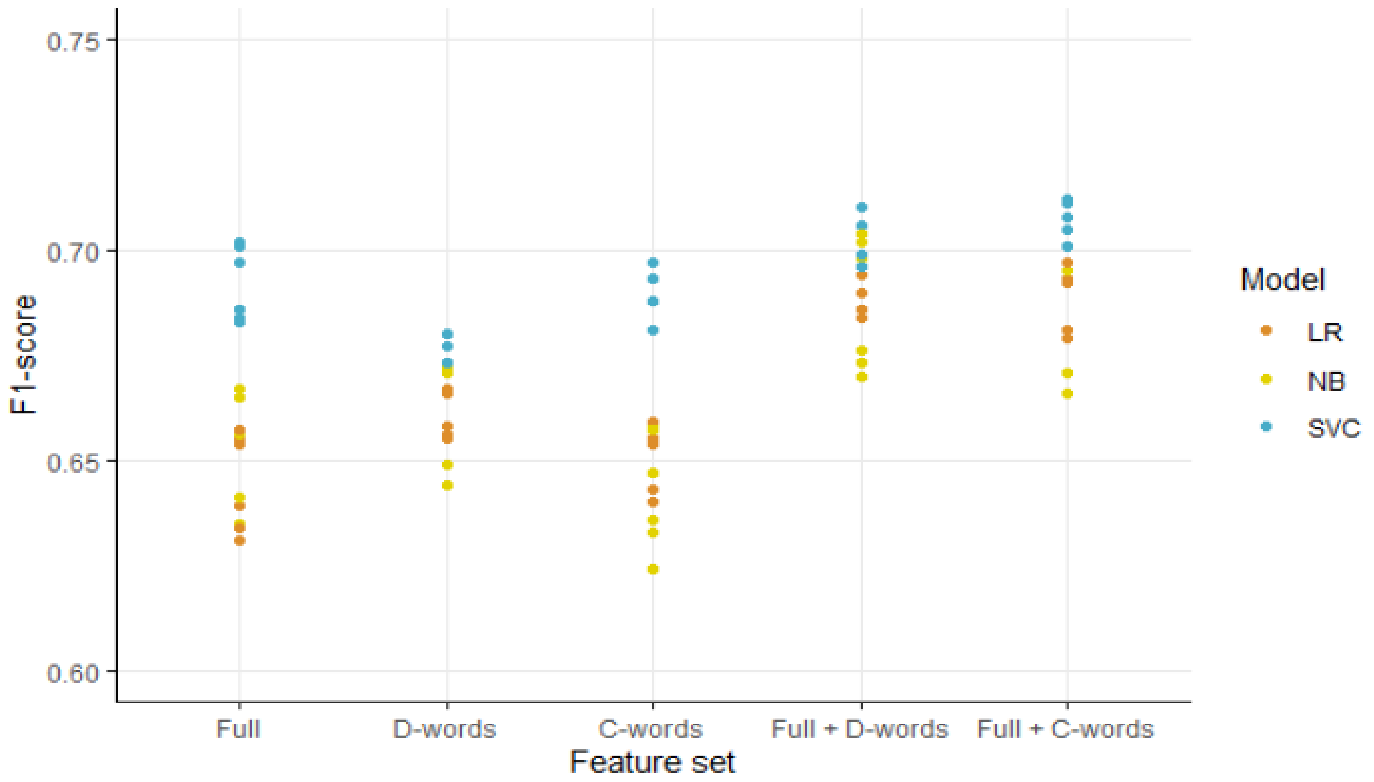
Experiments: different feature sets, classifiers, pre-processing steps , direct v.s. hierarchical classification

Performance: F1, accuracy, MCC score, macro-average weighted by # URLs per NACE code.

Web Intelligence Network

Funded by the European Union

Case Study



0.75
0.70
0.65
0.60
F1-score
Full
D-words
C-words
Full + D-words
Full + C-words
Feature set
Model
LR
NB
SVC

Case Study

Main results:

- Differences among pre-processing settings were very small (not shown)
- The support vector machine models best
- Hierarchical classification performed slightly worse than the direct classification
- Limited effect of feature types but full + D-words & Full + C-words performed best.
- Best model had a weighted F1 score of 0.712 (top 1 prediction) and 0.849 (top3 prediction)

Funded by the European Union

NACE Code Classification

Johannes Gussenbauer, Manveer Mangat, Alexander Kowarik
Statistics Austria

Web Intelligence Network

Funded by the European Union

Case Study Statistics Austria

Aim: Predict main activity of legal units

- Eventually use predictions to help with editing NACE codes in BR

Data: URL pairs found while scraping for ICT Survey (2019 – 2021)

- Deterministic URL-linking
- Models trained on ICT Survey (2019-2021) – results presented for ICT Survey 2021

Funded by the European Union

Pre-processing of scraped text

- Text on the landing page and sub-pages containing certain key-words in the link are scraped

- Only text elements are kept and further processed (removal of digits and punctuations, removal/replacement of characters not part of the German dictionary, etc)

- Currently apply
  1. “German morphological lexicon” (http://www.danielnaber.de/morphologie/)
  2. Lemmetization
  3. Stemming

Funded by the European Union

Feature selection:

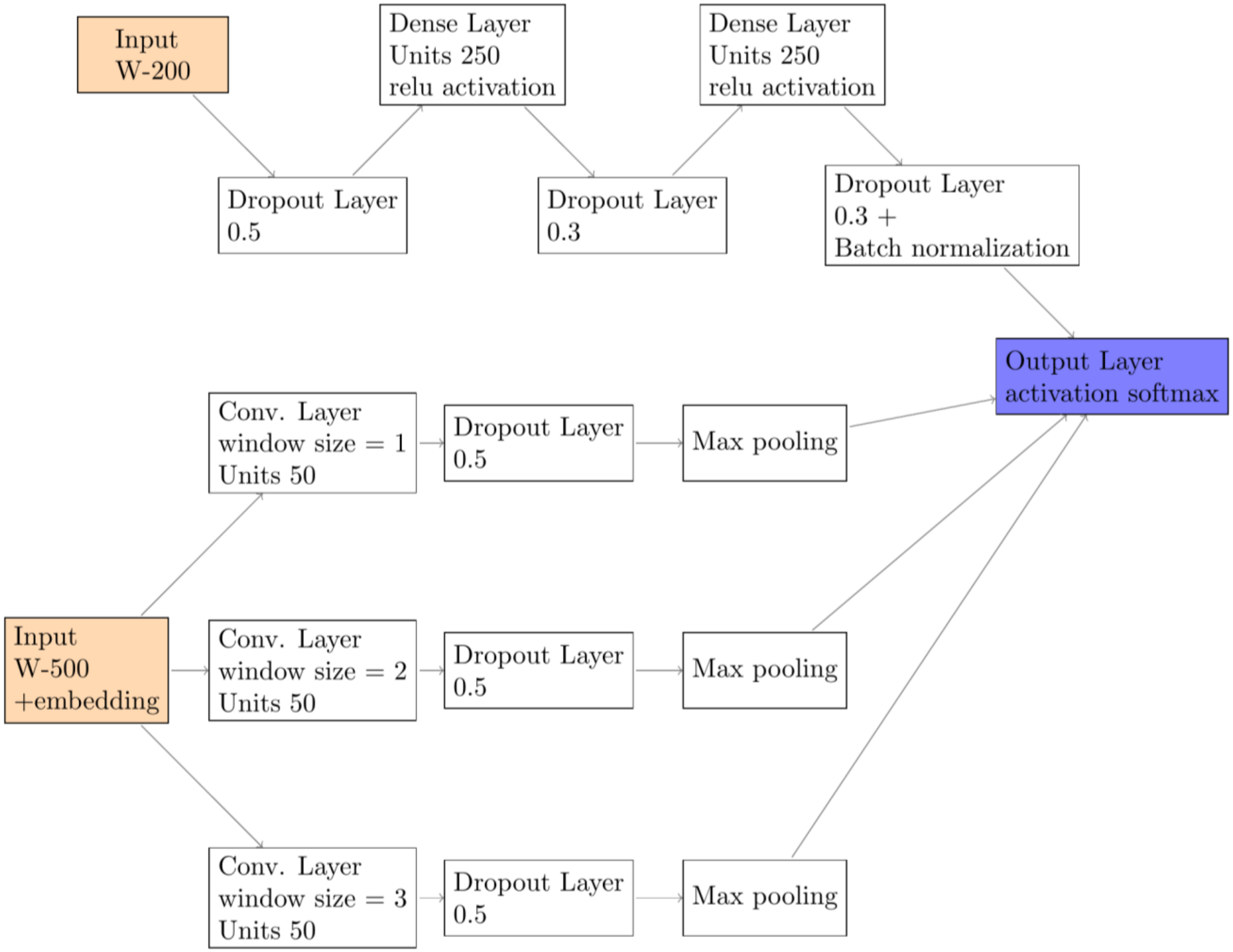
- After pre-processing scraped text contains >1Mio distinct words
- Idea: use the words and descriptions for NACE classification used by STAT (~ 20 000 words) as features→ Problem: 34% of these words did not appear in our web scraped texts
- Solution: combine a global and a local feature selection score function to select a balanced set of features (“An Improved Global Feature Selection Scheme for Text Classification.” Uysal (2016))
- selection strategy is applied to all the training data to select 200 and 500 words for each NACE2 code, W-200, W-500, respectively

Model specification

- Model 1:
  consists of feedforward layers and has as input the one-hot encoded W-200 words from the webpages weighted by the term frequency-inverse document frequency transformation (*Wide*)

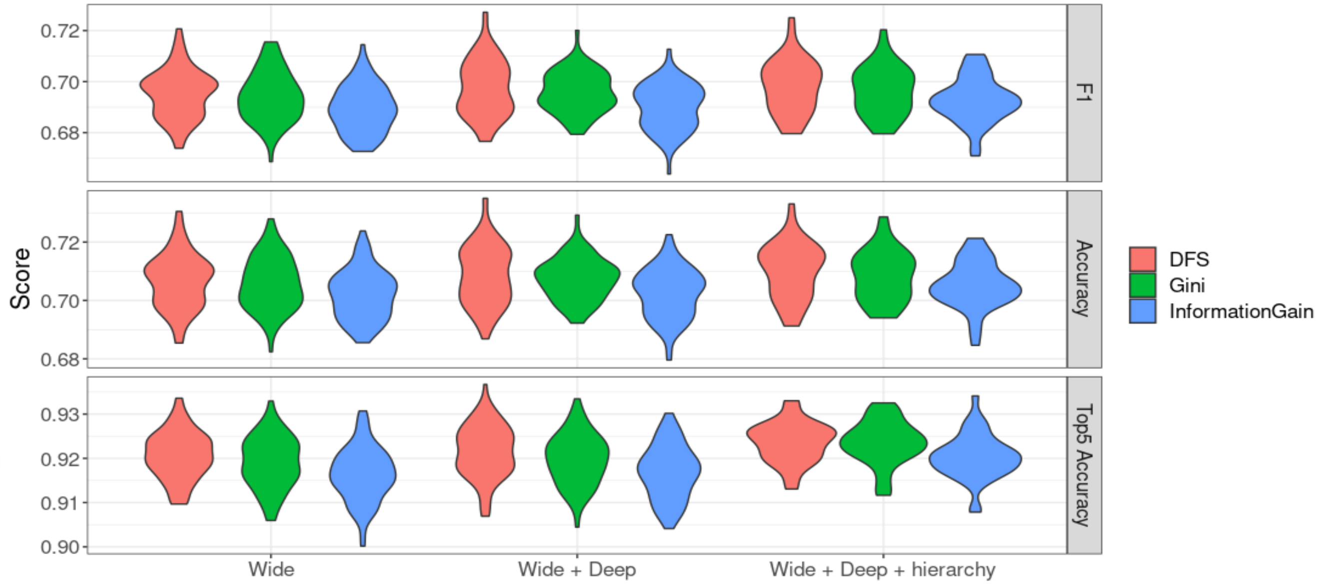
- Model 2: consists of the first one with an additional structure (*Wide + Deep*):
  a) W-500 transformed using pre-trained word embeddings from fastText
  b) additional structure consists of multiple convolutional filters applied to the word embeddings
  c) results from the feed forward and convolutional layers are concatenated in an penultimate layer
  d) then supplied to a final softmax layer
  e) R-Package keras, see Allaire and Chollet (2019), and the tensorflow software Abadi et al. (2015) used

- Model 3 (*Wide + Deep +Hierarchy*):
  - refers to applying the cross-validation first for predicting the NACE 1 level and using the predicted probabilities for the NACE 1 category as predictors for predicting the NACE 2 level


Input
W-200
Dense Layer
Units 250
relu activation
Dense Layer
Units 250
relu activation
Dropout Layer
0.5
Dropout Layer
0.3
Dropout Layer
0.3 +
Batch normalization
Output Layer
activation softmax
Conv. Layer
window size = 1
Units 50
Dropout Layer
0.5
Max pooling
Input
W-500
+embedding
Conv. Layer
window size = 2
Units 50
Dropout Layer
0.5
Max pooling
Conv. Layer
window size = 3
Units 50
Dropout Layer
0.5
Max pooling

Results

- 40 cross-validation runs: training (80%), validation (10%), test (10%)



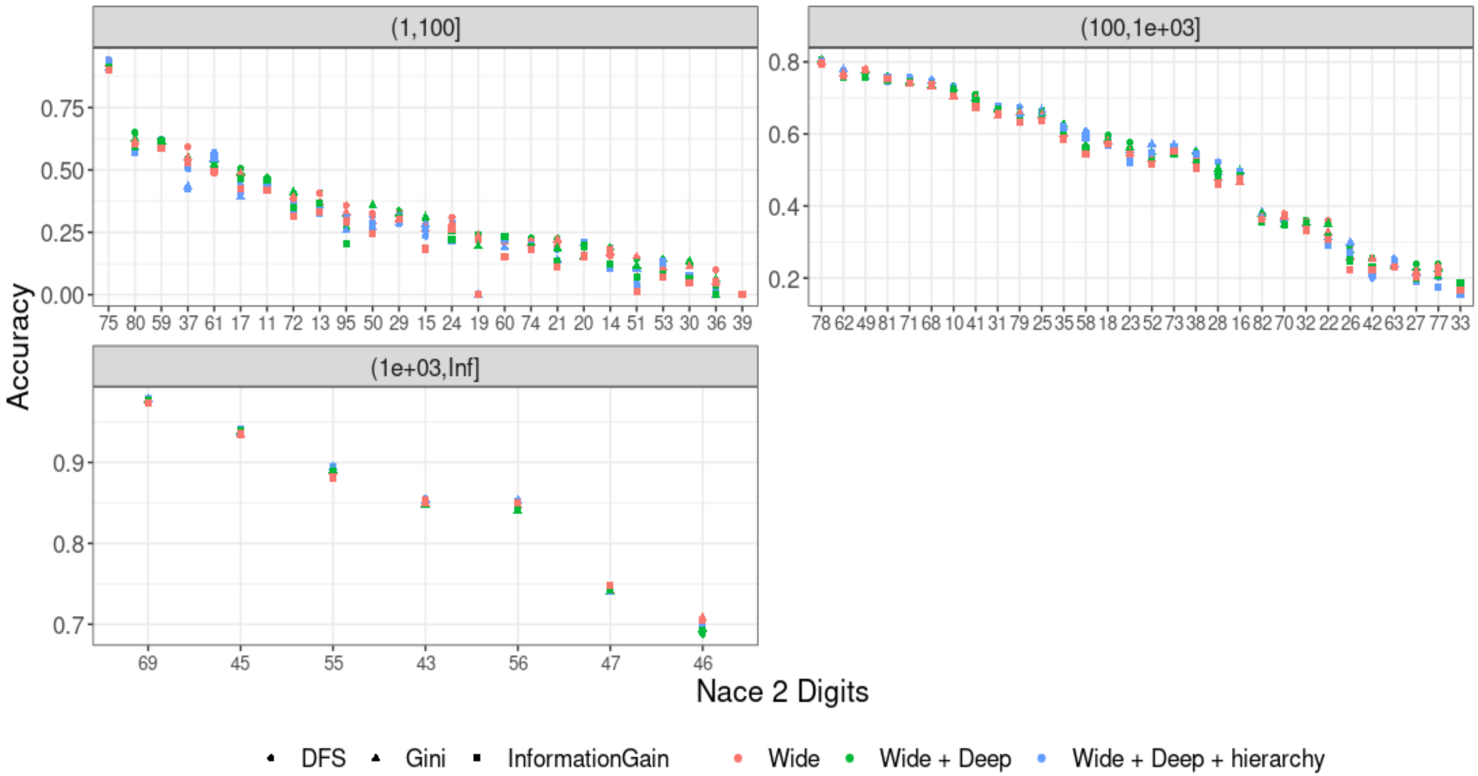
→ hardly any differences between the model settings and feature selection score

Web Intelligence Network

Funded by the European Union

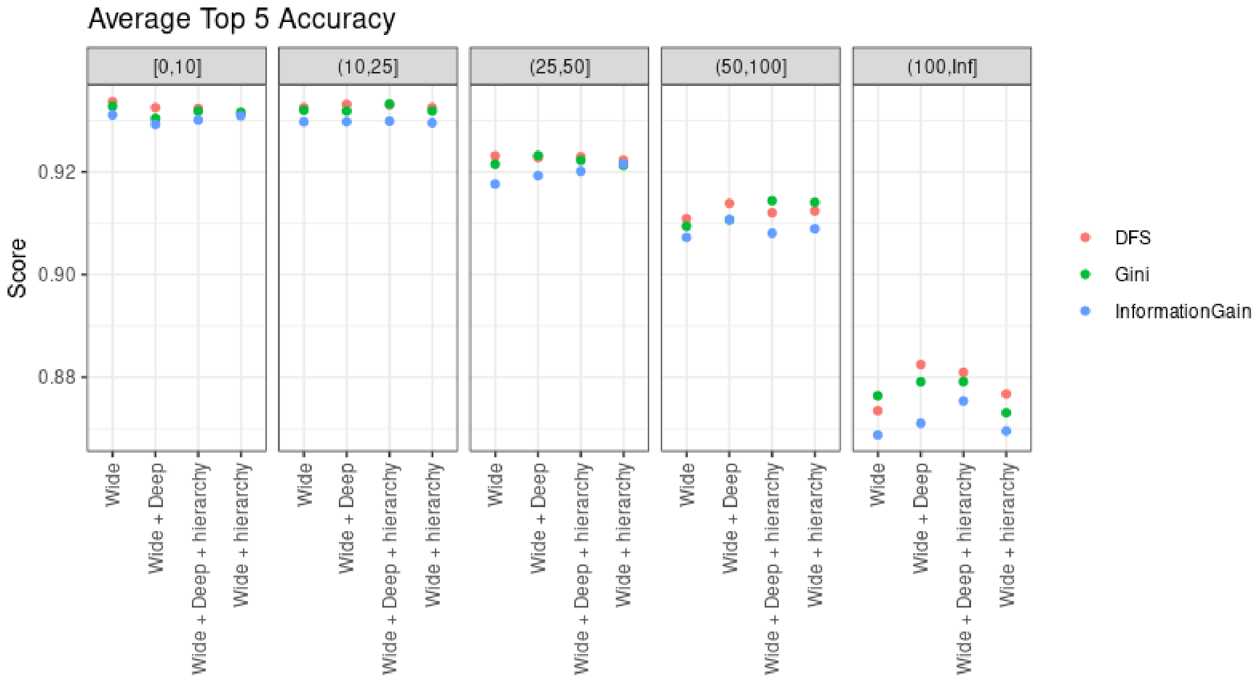
Results

- Average accuracy (y-axis) by NACE 2 digits (codes) (x-axis) for each model specification and feature selection score. The panels split the NACE 2 codes by number of enterprises available in the training data:



Results

- Average accuracy (y-axis) by company size (~employed persons) for each model specification and feature selection score.



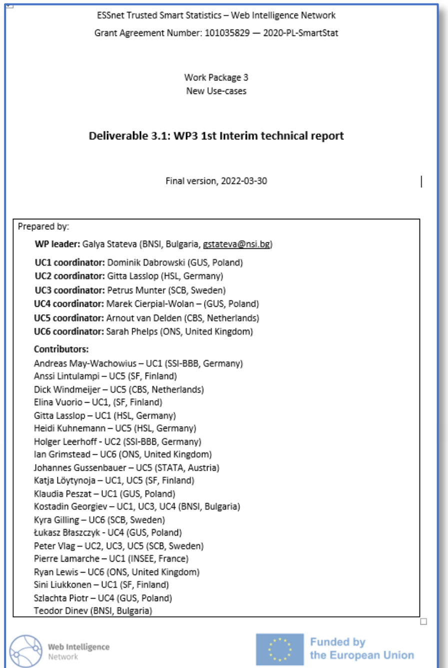
Web Intelligence Network

Funded by the European Union

More in depth reading

- Report: URL finding methodology (public on Cros portal)

- Deliverable 3.1: WP3 1st Interim technical report (internal)


ESSnet Trusted Smart Statistics – Web Intelligence Network
Grant Agreement Number: 101035829 — 2020-PL-SmartStat

Work Package 3
New Use-cases

Deliverable 3.1: WP3 1st Interim technical report

Final version, 2022-03-30

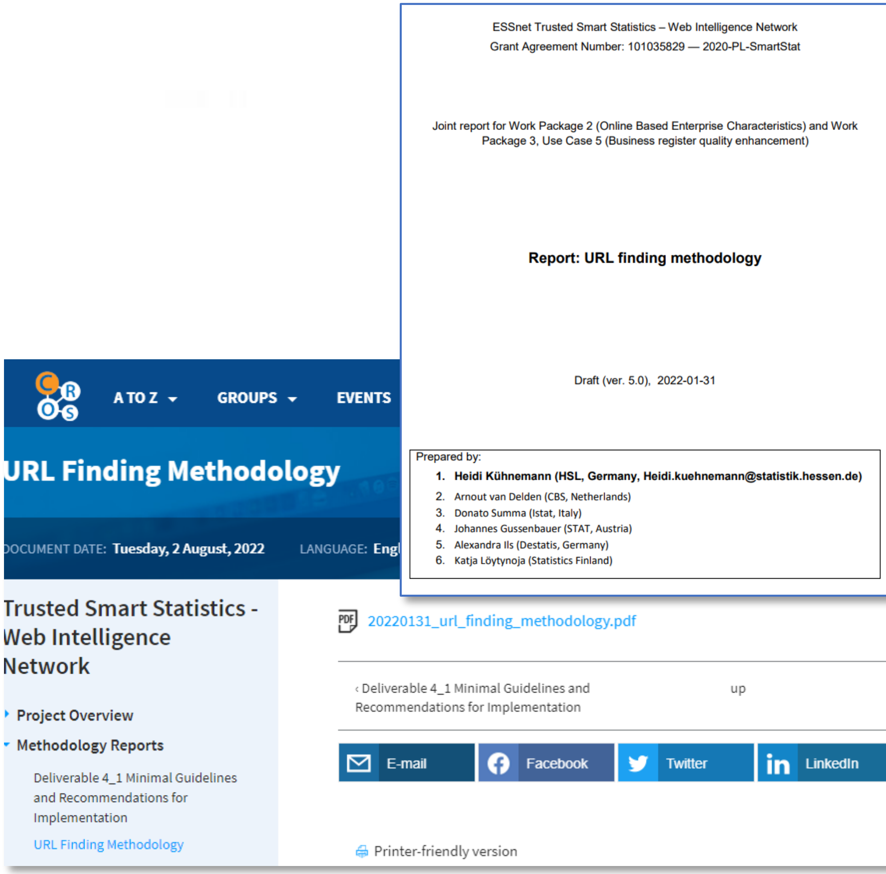
Prepared by:

WP leader: Galya Stateva (BNSI, Bulgaria, gstateva@nsi.bg)

UC1 coordinator: Dominik Dabrowski (GUS, Poland)
UC2 coordinator: Gitta Lasslop (HSL, Germany)
UC3 coordinator: Petrus Munter (SCB, Sweden)
UC4 coordinator: Marek Cierpial-Wolan – (GUS, Poland)
UC5 coordinator: Arnout van Delden (CBS, Netherlands)
UC6 coordinator: Sarah Phelps (ONS, United Kingdom)

Contributors:
Andreas May-Wachowius – UC1 (SSI-BBB, Germany)
Anssi Lintulampi – UC5 (SF, Finland)
Dick Windmeijer – UC5 (CBS, Netherlands)
Elina Vuorio – UC1, (SF, Finland)
Gitta Lasslop – UC1 (HSL, Germany)
Heidi Kuhnemann – UC5 (HSL, Germany)
Holger Leerhoff - UC2 (SSI-BBB, Germany)
Ian Grimstead – UC6 (ONS, United Kingdom)
Johannes Gussenbauer – UC5 (STATA, Austria)
Katja Löytynoja – UC1, UC5 (SF, Finland)
Klaudia Peszat – UC1 (GUS, Poland)
Kostadin Georgiev – UC1, UC3, UC4 (BNSI, Bulgaria)
Kyra Gilling – UC6 (SCB, Sweden)
Łukasz Błaszczyk - UC4 (GUS, Poland)
Peter Vlag – UC2, UC3, UC5 (SCB, Sweden)
Pierre Lamarche – UC1 (INSEE, France)
Ryan Lewis – UC6 (ONS, United Kingdom)
Sini Liukkonen – UC1 (SF, Finland)
Szlachta Piotr – UC4 (GUS, Poland)
Teodor Dinev (BNSI, Bulgaria)

Web Intelligence Network
Funded by the European Union


ESSnet Trusted Smart Statistics – Web Intelligence Network
Grant Agreement Number: 101035829 — 2020-PL-SmartStat

Joint report for Work Package 2 (Online Based Enterprise Characteristics) and Work Package 3, Use Case 5 (Business register quality enhancement)

Report: URL finding methodology

Draft (ver. 5.0), 2022-01-31

Prepared by:
1. **Heidi Kühnemann (HSL, Germany, Heidi.kuehnemann@statistik.hessen.de)**
2. Arnout van Delden (CBS, Netherlands)
3. Donato Summa (Istat, Italy)
4. Johannes Gussenbauer (STAT, Austria)
5. Alexandra Ils (Destatis, Germany)
6. Katja Löytynoja (Statistics Finland)

https://ec.europa.eu/eurostat/cros/content/url-finding-methodology_en

Lessons Learned

Arnout van Delden, Johannes Gussenbauer

Web Intelligence Network

Funded by the European Union

Train- test set construction

Issues:

- Not always a 1:1 link between website and enterprise
- Enterprises often have multiple activities: predict more labels
- Difficult to obtain error-free training material
- Website texts over report certain activities (sales, quality) and under report others (production)

Some options to deal with the issues:

- Drop the uncertain cases from the train-test set
- Use a large train set
- Use a more robust ML algorithm to deal with noise

Features

- Texts from which part of the website?
  - Landing page, about us page, contact page.

- Feature derivation – how to get rid of (some) noise? Points to consider:
  - Knowledge-intensive or not?
  - Context or not?
  - Language-specific standardisation
  - Different phrasing on websites than in NACE classification definitions

- Properly processing inputs can be more important than choice of algorithm (“rubbish in rubbish out”)

Algorithms

- Some models have large difference between train and test performance: check for overfitting in the CV procedure
- Confidences can be calibrated into probabilities: a good calibration set is needed
- Splitting data and training multiple models also makes sense when putting procedure in production -> smooth predictions

Web Intelligence Network
Funded by the European Union

Classes to predict

- High level NACE codes are heterogeneous: more training examples
- Class ‘Other’ very difficult to predict
- Rare classes: less training material and also not so interesting to automate

Options:

- Predict in different rounds? From more to less certain/ easy classes
- Skip rare and more difficult classes

Performance scores

- Think carefully what you want to achieve
  - Automatic coding / generating predictions / derive estimates from predicted probabilities/ …

- When every NACE code is equally important, use a macro score

- If you predict multiple labels for a website and only one has to be true, then adjust your performance measure to that situation

- Can be very useful to study which errors the model makes and to which factors they relate

Web Intelligence Network

Funded by the European Union