RQ1, ontology analysis

Wouter Van Rossem

2020/10/01

Outline

1. Work done for coding and organising the atlas.ti project.
2. Modifications and new features of the ontology analysis tool.
3. Outline of methods and data that can be used to answer the research questions.

Data analysis (1)

To do the actual data analysis:

- We reviewed the different proposed methods, questions and hypotheses that were previously formulated.
- For each of these we tried to identify methods and data that can be used to answer the question.
- In many cases this led us to introduce additional specifications of code groups.

Data analysis (2)

A note on the data used in the following slides. In general the process was a follows:

- Queries were executed in the tool, e.g. intersection of SIS and VIS.
- The output data generated through the graph analysis was downloaded as CSV files.
- These CSV files are directly loaded & processed in generating this presentation.

```
diff <- read.csv(file="../data/eurodac_particular.csv",head=TRUE,sep=",")
```

- Not all output of this kind of code will be shown in the following slides.

Question 1

> For each system, which categories are peculiar/native?

- **Method**: Using *set difference* of presence of categories of document group compared to a selection of document groups.
- **Example**: A comparison of categories native to each EU system, in relation to the other EU information systems.

1. For each system

- $Presence(Eurodac) \setminus (Presence(SIS) \bigcup Presence(VIS))$
- $Presence(SIS) \setminus (Presence(Eurodac) \bigcup Presence(VIS))$
- $Presence(VIS) \setminus (Presence(SIS) \bigcup Presence(Eurodac))$

2. Download CSV files

```
diff_eurodac <- read.csv(file="../data/eurodac_particular.csv",head=TRUE,sep=",")
diff_eurodac <- diff_eurodac[diff_eurodac$node_type == "code-group",]
diff_eurodac <- diff_eurodac$node_name
```

3. Combined results in table

```
max.len = max(length(diff_eurodac), length(diff_sis), length(diff_vis))
combined <- data.frame(
  eurodac=c(diff_eurodac, rep("", max.len - length(diff_eurodac))),
  sis=c(diff_sis, rep("", max.len - length(diff_sis))),
  vis=c(diff_vis, rep("", max.len - length(diff_vis))))
knitr::kable(combined)
```

eurodac
sis
vis
asylum
additional info / comments
date of application
date of exit
criminal offence data
education
linking data: MS
occupation data
operator data
occupation data: current
travel: relocation
parents data
personal ties
personal ties: in EU
residence
stay data
travel data
travel document data
travel document: validity: expiration
travel document: visa-related data

Question 2

For each system, what is the link between the applicant and Europe?

- **Method**: check the presence of a selection of groups code groups that represent links with Europe, such as:
  - `personal ties: in EU`
  - `migrant type` and `migration type`: "family reunification"
  - (future) `occupation` => `employment-interest` (IOM) / "skills"?
  - education
  - family reunification
  - residence
  - previous link with EU: `date of exit`, `return decision`
  - temporary visa & stay
  - …?
- To get more detailed types of links, we decided to further specify some code groups:
  - To distinguish between short, longer, and previous links to Europe.
  - And to distinguish between categories such as current employment vs possible future occupation.
- **Example**: For each EU systems we take the groups of categories present, and filter on the previous selection of code groups.

1. Check data for Eurodac, SIS and VIS & filter on groups

```
groups <- c("residence", "education", "date of exit", "personal ties: in EU", "migrant type", "travel documen

links_eurodac <- read.csv(file="../data/eurodac.csv",head=TRUE,sep=",")
links_eurodac <- links_eurodac[links_eurodac$node_name %in% groups,]
links_eurodac <- links_eurodac$node_name
```

2. Combine results into a table

```
max.len = max(length(links_eurodac), length(links_sis), length(links_vis))
combined <- data.frame(
  eurodac=c(links_eurodac, rep("", max.len - length(links_eurodac))),
  sis=c(links_sis, rep("", max.len - length(links_sis))),
  vis=c(links_vis, rep("", max.len - length(links_vis))))
knitr::kable(combined)
```

eurodac
sis
vis
date of exit
residence
linking data: MS
education

occupation data: current
personal ties: in EU
residence
stay data
travel document: visa-related data

Question 3

How is the division of labour between EU and MSs, and among MSs architected, with the mediation of IOs?

- **Method**
  - Comparison of code groups: medical data (+vulnerability), relocation data, asylum, housing, ...
  - Do we need to introduce more fine grained code group type for labour type?
- **Ideas**
  - How can we use linking data, e.g. to give indication of kinds of collaboration?
  - Intermediary (same category) vs mediator (different).
  - Can we use extra information from fieldwork?
- **Example** For each EU systems we take the groups of categories present, and filter on the selection of code groups representing types of labour.

1. Check data for Eurodac, SIS, VIS (and XKA) & filter on groups

```
groups <- c("medical data", "vulnerability", "travel: relocation", "asylum", "Temporary accomodation/housing'

links_eurodac <- read.csv(file="../data/eurodac.csv",head=TRUE,sep=",")
links_eurodac <- links_eurodac[links_eurodac$node_name %in% groups,]
links_eurodac <- links_eurodac$node_name
```

2. Combine results into a table

```
max.len = max(length(links_eurodac), length(links_sis), length(links_vis), length(links_xka))
combined <- data.frame(
  eurodac=c(links_eurodac, rep("", max.len - length(links_eurodac))),
  sis=c(links_sis, rep("", max.len - length(links_sis))),
  vis=c(links_vis, rep("", max.len - length(links_vis))),
  xka=c(links_xka, rep("", max.len - length(links_xka))))
knitr::kable(combined)
```

eurodac
sis

vis
xka
asylum
criminal offence data
Temporary accomodation/housing
travel: relocation
asylum
travel: relocation
vulnerability

Question 4

How does data from some actors become seen as more reliable?

- **Method**: using centrality: more connected nodes gives an indication of categories that are used.
- **Example**: (closeness) centrality of code groups in union of all code groups of Eurodac, SIS, & VIS:

```
ccn <- read.csv(file="../data/eurodac_sis_vis.csv",head=TRUE,sep=",")
ccn <- ccn[ccn$node_type == "code-group",]
ccn <- ccn[order(ccn$node_centrality_closeness),]
ccn <- ccn[order(ccn$node_centrality_closeness, decreasing=TRUE), c("node_name", "node_centrality_closeness"
knitr::kable(ccn)
```

node_name
node_centrality_closeness
15
System pre-existant
0.8545
14
System native
0.8439
17
biometrics
0.7487
20
linking data
0.7487
18
biometrics: fingerprint data
0.7169
23
procedure data
0.7008

24
sex & gender
0.6852
45
birth data
0.6847
51
name: surname
0.6847
49
name: earlier/other names
0.6741
50
name: forename
0.6635
52
nationality
0.6635
96
travel document data
0.6579
44
biometrics: photograph data
0.6529
47
birth data: place of birth
0.6529
95
travel data
0.6474
46
birth data: date of birth
0.6423
98
travel document: visa-related data
0.6368
86
date of application
0.6262
88
occupation data
0.6262
89
occupation data: current

0.6262
87
education
0.6156
90
parents data
0.6156
91
personal ties
0.6156
92
personal ties: in EU
0.6156
93
residence
0.6156
94
stay data
0.6156
97
travel document: validity: expiration
0.6156
16
asylum
0.5437
43
additional info / comments
0.5384
48
criminal offence data
0.5384
19
date of exit
0.5331
21
linking data: MS
0.5331
22
operator data
0.5225
25
travel: relocation
0.5225

Question 5

How does data become important to link different systems?

- **Method**: centrality: more connected nodes gives an indication of categories that can connect different data models.
- **Example**: betweenness centrality of code groups of Eurodac, SIS, & VIS:

```
bc <- read.csv(file="../data/eurodac_sis_vis.csv",head=TRUE,sep=",")
bc <- bc[bc$node_type == "code-group",]
bc <- bc[order(bc$node_centrality_betweenness),]
bc <- bc[order(bc$node_centrality_betweenness, decreasing=TRUE), c("node_name", "node_centrality_betweenness"
knitr::kable(bc)
```

	node_name	node_centrality_betweenness
14	System native	1215.7452
15	System pre-existant	947.1461
17	biometrics	523.0731
20	linking data	440.9926
18	biometrics: fingerprint data	290.8061
23	procedure data	238.5303
49	name: earlier/other names	195.4587
24	sex & gender	106.9462
51	name: surname	98.9926
50	name: forename	79.2792

52
nationality
69.1563
44
biometrics: photograph data
65.8985
45
birth data
42.7664
47
birth data: place of birth
23.0454
46
birth data: date of birth
15.1015
96
travel document data
6.6667
95
travel data
4.0000
16
asylum
2.0000
98
travel document: visa-related data
2.0000
86
date of application
0.5000
19
date of exit
0.4000
21
linking data: MS
0.4000
88
occupation data
0.4000
89
occupation data: current
0.4000
22
operator data

0.0000

25

travel: relocation

0.0000

43

additional info / comments

0.0000

48

criminal offence data

0.0000

87

education

0.0000

90

parents data

0.0000

91

personal ties

0.0000

92

personal ties: in EU

0.0000

93

residence

0.0000

94

stay data

0.0000

97

travel document: validity: expiration

0.0000

Open questions

- How are links between different actors mediated through EU, MS, and international organization systems?
- How do some actors become more important?
- How do different social-cultural context in which systems were created enact people differently?
- How are responsibilities between organisations architected?
- When can data modifications become unknown to other actors that reused the data?
- How do travel documents become important for identifying persons?
  - How are id documents and compared to biometrics more/less important

for connecting the systems

Questions and ideas

- Registration data vs application data
- When and for how long are identities stored
- Fylakio xls as disinscription, comparison with XKA and others