Detecting and Reporting Extensional Concept Drift in Statistical Linked Data

Meroño-Peñuela, A.; Guéret, C.; Hoekstra, R.; Schlobach, S.

Publication date
2013
Document Version
Final published version
Published in
SemStats 2013 : Semantic Statistics 2013

Detecting and Reporting Extensional Concept Drift in Statistical Linked Data

Albert Meroño-Peñuela[1,2], Christophe Guéret[2], Rinke Hoekstra[1,3], and Stefan Schlobach[1]

[1] Department of Computer Science, VU University Amsterdam, NL
albert.merono@vu.nl
[2] Data Archiving and Networked Services, KNAW, NL
[3] Leibniz Center for Law, Faculty of Law, University of Amsterdam, NL

Abstract. The RDF Data Cube vocabulary is a catalyst for the availability of statistical Linked Data: raw statistical Linked Data are easy to model in, publish to, and retrieve from the Linked Data cloud. In statistical datasets, *concepts* are central entities represented by variables and their values. The meaning of these concepts is often assumed to be stable, but in fact it can change over time: we call this *concept drift*. Extensional concept drift is one type of change of meaning that affects the things the concept extends to. It occurs frequently in historical datasets, and it can have drastic consequences on longitudinal querying. In this paper we propose and use a method to detect extensional concept drift in a dataset modelled using the RDF Data Cube vocabulary: the Dutch historical censuses. We analyze, model and publish back the occurrence of extensional concept drift in concepts of the occupation census, advocating straightforward publishing of results in a pull-push workflow.

Keywords: Concept Drift, Semantic Web, Statistical Linked Data

1 Introduction

Availability of statistical Linked Data is growing[4,5]. The RDF Data Cube vocabulary, now a W3C candidate recommendation, is a catalyst for statistical data exchange in the Semantic Web. It implements the hypercube model that underlies SDMX. Most datasets that use RDF Data Cube, like statistical data in general, assume some degree of stability in the concepts (variables, values) they refer to. But these concepts can change their meaning over time, especially when a wide time range is covered. In this paper we find and report back this change of meaning of concepts, or concept drift, in a historical census dataset. We identify the concepts that changed their meaning, and we upload these results to the Linked Data cloud to improve their reusability and reproducibility.

This paper proposes a solution for the problem of identifying concept drift. Concept drift refers to the change of meaning of concepts, as the reality they

[4] http://lod-cloud.net/state/
[5] http://eurostat.linked-statistics.org/

model changes continuously. Concept drift can have drastic consequences on the use of a concept in an application. A concept may replace the meaning of other concepts, or other concepts can take over its meaning. This produces errors in data retrieval that are very difficult to trace and address. Concept drift can happen at the concept identifier level (*label drift*), in the basic properties of the concept (*intensional drift*), or to the things the concept refers to (*extensional drift*) [22]. This paper proposes a statistics-based solution for the latter.

Other approaches propose solutions to solve concept drift (see Section 2), but they rely on *a)* availability of data about individuals, or *b)* some formalization (often OWL ontologies) of the concepts. In statistical data none of these may be available (see Section 4), and therefore our proposal exploits the statistical properties of quantifiable observations.

We apply our method to the Dutch historical censuses dataset (1795-1971). One fundamental problem with time series like these is backwards comparability. Several techniques have been developed to allow consistent comparison across versions, like classification schemes and mapping of concepts. In our case, year-dependent classification schemes and mappings of occupations with a historical standardized classification are available. We apply extensional drift detection to support these knowledge engineering tasks, finding that 42 out of 217 (19.35%) concepts suffer extensional concept drift.

This is useful for three different user communities. First, users without statistical skills will be aware of data anomalies without the need of running any concept drift detection method again. Second, social historians will gain insight on social dynamics of the past, as they may recognise drifted concepts that explain some historical reality. Third, generic users will benefit of concept drift aware applications, that will retrieve more reliable data considering these drifts.

Tracing the dynamics of meaning is not free of obstacles. First, since concept drift can take years to occur, data covering a large time range is necessary. Such historical datasets are often very messy and heterogeneous, and querying them successfully and reliably is not trivial. Second, finding an appropriate implementation for extensional drift is an arduous task because of the existence of multiple statistical tests covering a large variety of situations. To answer the first question, we propose a SPARQL query template for RDF Data Cube datasets. This template can be used to generate all the queries needed to retrieve statistical data. With respect to the second question, we study our source data distribution and we propose a statistical hypothesis that can be accepted or rejected using a statistical test.

This paper is organised as follows. In Section 2 we describe the state of the art in concept drift and statistical data publishing. In Section 3 we set the formal framework for the study of concept drift. In Section 4 we describe an experiment to detect and report extensional concept drift in the Dutch historical occupational census dataset. Finally, in Section 5 we establish some conclusions and further work.

2 Related Work

Concept drift is a very active research topic in Machine Learning [21], where it is defined as the situation in which the statistical properties of a target variable change over time in unforeseen ways. Learning from data streams in such a situation requires a concept drift detection method [1,4,8,17]. On the Semantic Web, concept drift relates to the study of the dynamics of meaning. In ontologies, this is addressed in ontology change and evolution management ([7,10,14]). Fanizzi et al. [5] propose a method based on clustering similar instances. But Description Logics have also addressed the related problem of detecting differences between ontologies [9]. To the best of our knowledge, the work of Wang et al. [22,15] is the only concept drift formalization in a Semantic Web setting.

Although concept drift is the central topic in this paper, we also consider important to mention contributions about statistical data publishing on the Web. SDMX[6] is the ISO standard for statistical data exchange. The representation of statistical data as Linked Data started with SCOVO [12] and continued with the RDF Data Cube vocabulary [3], which is SDMX compatible. Closely related to our use case, there is work on publishing statistical Linked Open Government Data [20], and concretely census data[7] [6,16,18].

3 Concept Drift

The world is continuously changing, and concepts also change over time. A concept refers to different objects, real or abstract, at different points in time. For instance, the concept `Manager` refers to different types of occupations in 1795 and in 2013. We use the formalisation framework described by Wang et al. [22] in order to address change of meaning over time.

Definition 1. *The* meaning of a concept *C is a triple (label(C),int(C),ext(C)), where label(C) is a string, int(C) a set of properties (the* intension *of C), and ext(C) a subset of the universe (the* extension *of C).*

All the elements of the meaning of a concept can change. To address concept identity over time, Wang et al. [22] assume that the intension of a concept C is the disjoint union of a rigid and a non-rigid set of properties (i.e. $(int_r(C) \cup int_{nr}(C))$). Then, a concept is uniquely identified by some essential properties that do not change. The notion of identity allows the comparison of two variants of a concept at different points in time, even if a change on its meaning occurs.

Definition 2. *Two concepts C_1 and C_2 are considered identical if and only if, their rigid intension are equivalent,* i.e.*, $int_r(C_1) = int_r(C_2)$.*

If two variants of a concept at two different times have the same meaning, there is no concept drift. We define intensional, extensional, and label similarity

[6] `http://www.sdmx.org/`

[7] See the US case, `http://www.rdfabout.com/demo/census/`

functions $sim_{int}, sim_{ext}, sim_{label}$ to quantify meaning similarity. Each of these functions has range $[0, 1]$, and a similarity value of 1 indicates equality.

Definition 3. *A concept has extensionally drifted in two of its variants C' and C", if and only if,* $sim_{ext}(C', C'') \neq 1$. *Intensional and label drift are defined similarly.*

To apply this framework of concept drift it is required to define intension, extension and labelling functions, and to define similarity functions over intension, extension and labels. We define these functions in Section 4.4.

4 Concept Drift in the Dutch Historical Censuses

Concept drift is a fundamental issue to be addressed in the Semantic Web, especially in historical datasets. In this section we describe the implementation of a method to identify and report extensional concept drift in a subset of the Dutch historical occupational censuses. Due to the time period the censuses cover (1795-1971) many concepts may have drifted [2]. We preprocess data in a typical data mining setting, and we detect extensional concept drift using standard statistical tools. Additionally, we propose and illustrate a pull-push statistical workflow that enriches the queried endpoint back with our results.

4.1 Linking and Publishing the Census

The Dutch historical censuses dataset[8] comprises 507 Excel workbooks containing 2,288 census tables. These tables have been created by hand, using the digitized images of the original census books. Census data refers here only to aggregated data (i.e. counts of people meeting certain conditions). Microdata (i.e. data about individuals) is not available in this dataset. Each census table describes a portion of the population, occupation or housing census of a certain year. In this experiment we use tables of the occupational census in the province of Noord-Holland in the years 1889 and 1899. Due to the late industrialization of the Netherlands in the 19th century, data of this time and region are inclined to contain more drastic changes in the occupational landscape.

Figure 1 shows the layout of one of these tables. In order to publish them as Linked Data, we implement a supervised Excel to RDF Converter: TabLinker[9]. TabLinker uses markup of cells to distinguish the different table elements: table name, data cells, column headers, row properties, row headers, hierarchical row headers, and metadata. Users need to manually style these table regions.

TabLinker uses a per-cell data model according to the RDF Data Cube vocabulary. We use the term *observation* to describe a data cell and its context (`d2s:isObservation a qb:Observation`). A data cell is linked to a number of dimensions (`d2s:dimension a qb:DimensionProperty`) that correspond to the

[8] `http://www.volkstellingen.nl`

[9] `https://github.com/Data2Semantics/TabLinker`

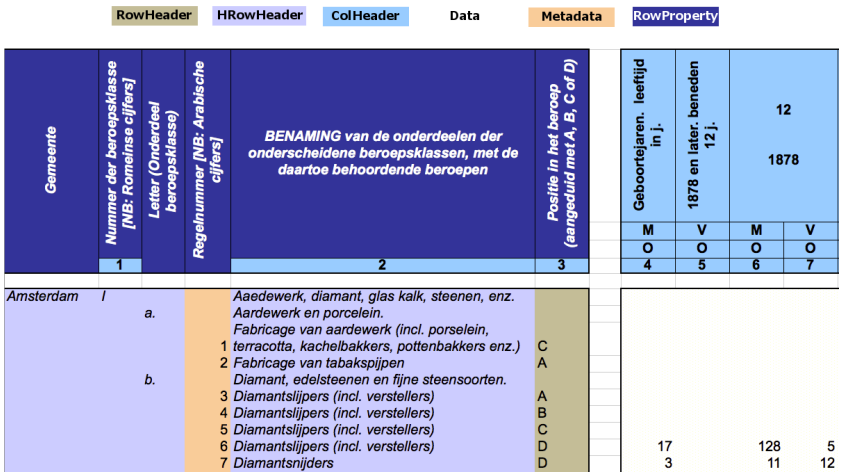

Fig. 1: Table of the occupational census of 1889, province of Noord-Holland. Legend illustrates identified cell regions (colours are merely indicative).

column and row headers of that cell. The content of a data cell is always a number counting population (`d2s:populationSize a qb:MeasureProperty`). TabLinker generates an interpretation of the table layout, reading table cell regions (see legend in Figure 1). *Data* cells contain the census counts. *ColHeaders* and *RowHeaders* contain classifications of age ranges and occupations, describing the numbers in their respective columns and rows. *RowProperties* link dimension nodes within an anonymous `qb:Observation` node, which is attached to a `d2s:Data` instance via a `d2s:isObservation` property. Cells marked as *HRowHeaders* contain hierarchical classifications, and TabLinker generates `skos:broader` triples for these conveniently. *Data* cells are linked to their corresponding *ColHeader* and *RowHeader* cells as Data Cube dimensions (`CellN28 d2s:dimension A, B`). One named graph is built per table.

4.2 Querying Cubes

After the table markup, we run TabLinker on the selected files, and we expose the generated named graphs in a SPARQL endpoint.[10]

Querying these graphs in an homogeneous way is challenging [16]. The tables from which they are generated are messy and inconsistent with respect to the layout (i.e. where things are located). They are extremelly sensitive to language changes (i.e. how things are labelled). In some cases, modelling and political decisions (e.g. in which group an individual has to be counted) also make comparisons difficult. To solve this, we design a SPARQL template that exploits

[10] `http://lod.cedar-project.nl:8080/sparql/cedar`

```
1  PREFIX qb: <http://purl.org/linked-data/cube#>
2  PREFIX d2s: <http://www.data2semantics.org/core/>
3  PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
4  PREFIX ns: <row_property_URI>
5
6  SELECT ?d1label ... ?dnlabel ?p1label ... ?pmlabel ?population
7  FROM <named_graph_URI>
8  WHERE {
9   ?cell d2s:isObservation [ a qb:Observation ;
10                            qb:DimensionProperty ?d1 ... ?dn ;
11                            ns:property1 ?p1 ;
12                            ...
13                            ns:propertym ?pm ;
14                            qb:MeasureProperty ?population ] .
15  OPTIONAL {
16  ?cell d2s:isObservation [ns:propertyk ?pk ] .
17  ?pk skos:prefLabel ?pklabel . }
18  ...
19  OPTIONAL {
20  ?cell d2s:isObservation [qb:DimensionProperty ?dl ] .
21  ?dl skos:prefLabel ?dllabel . }
22  ?pt skos:broader ?pu .
23  ?pu skos:broader ?pv .
24  d1 ... dn skos:prefLabel ?d1label ... ?dnlabel .
25  p1 ... pm skos:prefLabel ?p1label ... ?pmlabel .
26  FILTER (?d1 IN (v1, ..., vr)) ...
27  FILTER (?dn IN (w1, ..., ws))
28 }
```

Listing 1.1: SPARQL template for homogeneous querying. `qb:DimensionProperty` and `qb:MeasureProperty` can be replaced by other predicates of the same type.

the different cell regions (see Section 4.1) and accommodates a generic querying situation.

This query template is shown in Listing 1.1. To fill the template and generate valid SPARQL, first the user must manually choose which variables to query (line 6). Second, other more trivial queries aid the user to fill in the row property predicates ($property_1 ... property_m$, lines 11-13). Third, optional variables must be enclosed in `OPTIONAL` graph patterns (lines 15-21). Fourth, hierarchical properties must be traversed if necessary (lines 22-23). Fifth, labels need to be retrieved to avoid the presence of URIs in the resultset (lines 24-25). Finally, valid values for selected dimension variables need to be filtered out (lines 26-27). Some other queries help the user to assign which values belong to which variables.

We follow this procedure to generate the queries needed for our case.[11] As a result, we get translations of RDF Data Cube into very redundant resultsets that can be mined in a statistical environment. In this experiment we use R [19].

[11] `https://raw.github.com/albertmeronyo/ConceptDrift/master/sparql/queries.txt`

4.3 Preprocessing Census Data

We use the method presented in Section 4.2 to query RDF Data Cube census data and get them into R via the SPARQL R package [11]. We select the variables age range, position, gender, marital status, municipality, occupation and population for the two years. A sample of the retrieved data frames is shown in Table 1.

Age range	Position	Gender	M. status	Municipality	Occupation	Population
36-50	A	M	G	Velsen	Aanemers	3
51-60	B	V	O	Zaandam	Agenten	1
23-35	C	VROUWEN	O.	Haarlem	Ambtenaren en beambten	2
36-50	A	MANNEN	G.	Weesp	Afwerken van huizen	5

Table 1: Sample rows of the data frame returned by the SPARQL queries. The first two belong to the first dataset, last two to the second. Position stands for an occupational rank (*A* indicates directors or business owners, *C* ordinary workers). *M, MANNEN* stand for *men*; *V, VROUWEN* for *women*; and *G*, *O* for *married* and *unmarried*, respectively. *Aanemers* are *contractors*, *agenten* are *manufacturer's agents*, *ambtenaren en beambten* are *civil workers* and *afwerken van huizen* are *house finishers*.

However, data still present the issue of non normalized values. Some variable values may not be comparable by design, like age ranges (e.g. 21-26 and 26-31 versus 21-23 and 24-31), although in our constrained data age ranges are totally compatible. Normalization of factor labels in comparable variables is solved by replacing the original values with standard ones. This is the case for the variables gender (`M/V` and `MANNEN/VROUWEN` are replaced by `Male/Female`) and marital status (`G/O` and `G./O.` by `Married/Unmarried`). The variable position is already normalized in the raw data (values `A/B/C/D`).

We use external data sources to normalize complex variables that radically changed between time periods, like municipality and occupation. Concretely, we rely on existing mappings between occupation labels and unique identifiers of the Historical International Standard Classification of Occupations (HISCO).[12] These mappings are manually established by experts, pairing each occupation appearance in the census tables with one (and only one) HISCO code.[13] HISCO codes follow a tree-like structure: code `12310`, for instance, refers to occupational titles under the micro group `12310` (*Notary*), unit group `123` (*Notaries*), minor group `12` (*Jurists*), and major group `1` (*Professional, technical and related workers*). Only mappings with micro groups (i.e. five number codes), which are the leaves of the tree, are allowed.

Finally, we perform some cleaning, removing incomplete rows, partial total population aggregations, aggregations at the province level, and aggregations of smaller villages that only appear in one of the two datasets.

[12] `http://historyofwork.iisg.nl/`

[13] See `https://github.com/albertmeronyo/ConceptDrift/tree/master/stats`

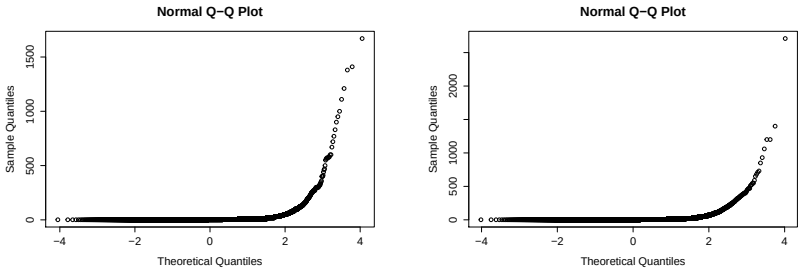

Fig. 2: Normal QQ-plots of all population counts of the 1889 and 1899 occupation tables of Noord-Holland. Both plots reveal non-normality of their distributions.

4.4 Extensional Concept Drift

We are interested in identifying extensionally drifted concepts in the census, that is, $sim_{ext}(C', C'') \neq 1$ for two given variants C', C'' of a concept C (see Section 3). Intuitively, this means that the instances of C have changed significantly.

We interpret extensional concept drift in a statistical setting. We define the extension function $ext(C)$ as the function that returns the number of individuals that belong to C, and the extension similarity function $sim_{ext}(C', C'')$ as the function that returns the probablity that C' and C'' have identical populations. Hence, we assume that the extension of C has drifted between C' and C'' iff the populations of C' and C'' are non identical (there is a shift between the populations of C' and C'') (see Section 3).

Using the data of Section 4.3, we want to know if population counts of people having the same occupations in Noord-Holland have identical data distributions between the years 1889 and 1899. Without assuming the data to have normal distribution (see Figure 2), we want to test at .05 significance level if the population counts for a given occupation have identical data distributions.

The null hypothesis H_0 is that the population counts of the two sample years are identical populations. To test the hypothesis, we run the Wilcoxon signed-rank test that comes with the R distribution [23]. Since the Wilcoxon test is symmetric, we assume $sim_{ext}(C', C'') = sim_{ext}(C'', C')$.

For example, the occupation *Contractors*, which has been normalized in both datasets with the HISCO code `21240` (see Section 4.3), has the population data arrays shown in Listing 1.2. We run the `wilcox.test` function using these arrays, concluding that the population of contractors in Noord-Holland in the years 1889 and 1899 are statistically identical populations ($p > 0.05, N = 42$, Wilcoxon signed-rank test). Consequently, there is no extensional drift in this case.

In order to have a complete overview on what occupational concepts drifted in this period, we extract all common HISCO codes in both datasets and iteratively run the Wilcoxon test on their population data. The first and second dataset have 57 and 88 non-common HISCO codes, respectivelly. 42 out of the 217 (19.35%) common HISCO codes are found to have p-values under the 0.05 threshold,

```
> x <- df1889[df1889.hisco$HISCO == '21240','population']
 [1]  3  1  1  2  2  1  3  1  1 10 10  5  2  1  1  1  1  4  1  4  1  1
 1  2  1  8  2  1  2  5  1 50  1  1  1  1  1  1  2  1
[41]  1  1
> y <- df1899[df1899.hisco$HISCO == '21240','population']
 [1] 20  3  1  3  1 30 10  5  4  1  4 10  1  9  1  3  1  8  1  4  4  1
 1  2  1  1  1  3  1  1  1  1  1  1  1  1  1  2  1  1
[41]  1  2
> wilcox.test(x, y)

        Wilcoxon rank sum test with continuity correction

data:  x and y
W = 830, p-value = 0.6063
alternative hypothesis: true location shift is not equal to 0
```

Listing 1.2: Arrays of population counts in 1889 and 1899 in Noord-Holland of HISCO code `21240` (*contractors*).

resulting in rejection of H_0 and consequently denoting extensional concept drift. Table 2 shows extensionally drifted and stable codes and major groups.

4.5 Reporting Back

The statistical linked data workflow we have followed works only one way. In Section 4.2 we connect the endpoint and the statistical environment online. In Section 4.4, we execute the drift method task offline, and chances are that the results are kept offline too. This generates a pull-not-push workflow, where data consumers are limited to only read SPARQL endpoints. Further queries will not retrieve these results, lowering their reusability and reproducibility. We propose to provide straightforward updates to the endpoint to push results. In particular, our recommendation is that *every statistical analysis on the Web should start with a SPARQL SELECT and end up with one (or more) SPARQL UPDATEs.*

We want users to know that some occupational concepts extensionally drifted between two time periods. Since the different variants of these concepts and the time they belong to are encoded in their URIs, we link these with an appropriate predicate and assign it a weight. We define this weight as the previously computed *p-value* between the two variants (see Section 4.4). We build a SPARQL UPDATE query (see Listing 1.3) iterating over all common HISCO codes URIs. We execute it against the same endpoint and named graph the original data were retrieved from.

5 Conclusions and Further Work

In this paper we present a workflow to detect and report extensional concept drift in statistical Linked Data, raising the importance of concept drift in statistical

HISCO	Occupation	p-value
97125	Loader of ship, truck, wagon or airplane	1.83e-10
21110	General manager	4.23e-09
41025	Working proprietor (wholesale, retail trade)	1.52e-08
79100	Tailor	7.75e-07
57030	Barber, hairdresser	1.17e-04
88010	Jeweller	1.84e-04

(a) Occupations with stronger ext. drift.

HISCO	Occupation	p-value
53190	Other cooks	1.00
75452	Lace weaver	1.00
75490	Other weavers	1.00
75990	Other spinners, weavers, knitters, dyers	1.00
77690	Other bakers, pastry cooks and confectionery makers	1.00

(b) Occupations with greater ext. stability.

Group	Type	p-value
7, 8, 9	Production, transport, operators	2.03e-19
5	Service workers	1.88e-12
4	Sales workers	2.94e-08
2	Administrative and managerial	4.20e-08

(c) Major groups with stronger ext. drift.

Group	Type	p-value
6	Agriculture, animal husbandry, fishermen, hunters	0.38
0, 1	Professional and technical	0.16
3	Clerical	1.40e-04

(d) Major groups with greater ext. stability.

Table 2: Wilcoxon test p-values per HISCO code ((a),(b)) and major group ((c),(d)).

analyses. The study of the dynamics of meaning in variables and values is critical, because uncontrolled concept drift produces wrong results in queries. In our use case, about one fifth of all analysed concepts present extensional concept drift.

These results are consistent with the slow industrialization of the Netherlands in the 19th century. Entrepreneurs put emphasis on trade rather than industry, although results show important variations in both groups. In this paper we only take extensional drift into account. Deciding to what extent this extensional drift implies a change on the meaning of a concept without a joint analysis of intension and labelling drift can be misleading. We address this below as future work.

We motivate the straightforward good practice of publishing such results back to the endpoint, closing a pull-push cycle. Non statistically versed Linked Data users will appreciate relevant statistical conclusions shared by others. Other domain experts, social historians in our case, will find out new metadata pushing forward their historical theses. Generic users will benefit from reusability and reproduciblity of results. All workflows in statistical Linked Data should begin with a SPARQL SELECT and finish with at least one SPARQL UPDATE.

We plan further work at several levels. First, we are working on releasing the HISCO normalization mappings we use as Linked Data, as well as an RDF version of HISCO. Additionally, we will make available a description of the `d2s` vocabulary as TabLinker matures, as well as a drift vocabulary when the study of its semantics becomes broader. Second, we plan to scale up the study of extensional drift in this dataset by including more census tables, minimizing user intervention on normalization of values by using embedded value mappings [13]. We will put special emphasis on how the method works depending on the

```
PREFIX d2s: <http://www.data2semantics.org/core/>
PREFIX d2s1889: <urn:nbn:nl:ui:13-m4k-4lp>
PREFIX d2s1899: <urn:nbn:nl:ui:13-988-0dq>

INSERT DATA {
  GRAPH <named_graph_URI> {
    d2s1889:Sjouwerlieden d2s:isDrift [
      d2s:extDrift d2s1899:Expeditie_bevrachters_bestellers_sjouwerlieden ,
                   d2s1899:Personeel_voor_laden_en_lossen ,
                   d2s1899:Personeel_voor_lading_en_lossing ,
                   d2s1899:Sjouwerlieden ;
      d2s:weight 1.83e-10 ] . } }
```

Listing 1.3: Excerpt of the SPARQL query reporting back extensionally drifted occupational concepts. Only the drift for one occupational concept is shown. Inverse drifts from the second graph to the first are also issued.

time gap between the data snapshots. We will study how the method works for other variables, like age ranges and municipalities. Finally, we aim at completing the study of concept drift by integrating intensional and labelling drift. We will leverage Linked Data to obtain additional knowledge about basic properties of these concepts and their linguistic characteristics.

Acknowledgements The work on which this paper is based has been partly supported by the Computational Humanities Programme of the Royal Netherlands Academy of Arts and Sciences, under the auspices of the CEDAR project. For further information, see `http://ehumanities.nl`. This work has been supported as well by the Dutch national program COMMIT.

References

1. Baena-García, M., del Campo-Ávila, J., Fidalgo, R., Bifet, A., Gavaldà, R., Morales-Bueno, R.: Early Drift Detection Method. In: Proc. ECML/PKDD 2006, Workshop on Knowledge Discovery from Data Streams. pp. 77–86 (2006)
2. Boonstra, O., Doorn, P., van Horik, M., van Maarseveen, J., Oudhof, J.: Twee eeuwen Nederland geteld. Onderzoek met de digitale Volks-, Beroeps- en Woningtellingen 1795–2001. DANS en CBS (2007)
3. Cyganiak, R., Reynolds, D., Tennison, J.: The RDF Data Cube Vocabulary. Tech. rep., World Wide Web Consortium (2013), `http://www.w3.org/TR/vocab-data-cube/`
4. Dries, A., Rückert, U.: Adaptive Concept Drift Detection. Statistical Analysis and Data Mining 2(5–6), 311–327 (2009)
5. Fanizzi, N., d'Amato, C., Esposito, F.: Conceptual Clustering: Concept Formation, Drift and Novelty Detection. In: The Semantic Web: Research and Applications, 5th European Semantic Web Conference. LNCS 5021. pp. 318–332. Springer (2008)
6. Fernández, J.D., Martínez-Prieto, M.A., Gutiérrez, C.: Publishing open statistical data: the Spanish census. In: Proceedings of the 12th Annual International Conference on Digital Government Research, DG.O 2011. pp. 20–25. ACM International Conference Proceeding Series (2011)
7. Flouris, G., Manakanatas, D., Kondylakis, H., Plexousakis, D., Antoniou, G.: Ontology change: classification and survey. The Knowledge Engineering Review 23(2), 117–152 (2008)

8. Gama, J., Medas, P., Castillo, G., Rodrigues, P.P.: Learning With Drift Detection. In: Advances in Artificial Intelligence - SBIA 2004, 17th Brazilian Symposium on Artificial Intelligence. LNCS 3171. vol. 3171, pp. 286–295. Springer (2004)
9. Gonçalves, R.S., Parsia, B., Sattler, U.: Analysing Multiple Versions of an Ontology : A Study of the NCI Thesaurus. In: Proceedings of the 24th International Workshop on Description Logics (DL 2011). vol. 745. CEUR Workshop Proceedings (2011), http://ceur-ws.org/Vol-745/
10. Gulla, J.A., Solskinnsbakk, G., Myrseth, P., Haderlein, V., Cerrato, O.: Semantic Drift in Ontologies. In: Proceedings of the 6th International Conference on Web Information Systems and Technologies. vol. 2. INSTICC Press (2010)
11. van Hage, W.R., with contributions from: Tomi Kauppinen, Graeler, B., Davis, C., Hoeksema, J., Ruttenberg, A., Bahls., D.: SPARQL: SPARQL client (2013), http://CRAN.R-project.org/package=SPARQL, R package version 1.15
12. Hausenblas, M., Halb, W., Raimond, Y., Feigenbaum, L., Ayers, D.: SCOVO: Using Statistics on the Web of Data. In: The Semantic Web: Research and Applications, 6th European Semantic Web Conference. LNCS 5554. vol. 5554, pp. 708–722. Springer (2009)
13. Jaiswal, A.: On Statistical Schema Matching with Embedded Value Mappings. Ph.D. thesis, The Pennsylvania State University (2012)
14. Klein, M.: Change Management for Distributed Ontologies. Ph.D. thesis, VU University Amsterdam (2004)
15. Meroño-Peñuela, A.: Semantic Web for the Humanities. In: The Semantic Web: Semantics and Big Data, 10th European Semantic Web Conference. LNCS 7882. pp. 645–649. Springer (2013)
16. Meroño-Peñuela, A., Ashkpour, A., Rietveld, L., Hoekstra, R., Schlobach, S.: Linked Humanities Data: The Next Frontier? A Case-study in Historical Census Data. In: Proceedings of the 2nd International Workshop on Linked Science (LISC2012). International Semantic Web Conference (ISWC). vol. 951. CEUR Workshop Proceedings (2012), http://ceur-ws.org/Vol-951/
17. Nishida, K., Yamauchi, K.: Detecting Concept Drift Using Statistical Testing. In: Discovery Science, 10th International Conference. Proceedings of DS 2007. LNCS 4755. vol. 4755, pp. 264–269. Springer (2007)
18. Petrou, I., Papastefanatos, G., Dalamagas, T.: Publishing Census as Linked Open Data. A Case Study. In: 2nd International Workshop on Open Data, WOD 2013 (2013), http://www-etis.ensea.fr/WOD2013/wp-content/uploads/2013/06/casestudyPetrou.pdf
19. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2013), http://www.R-project.org/
20. Salas, P.E.R., Martin, M., Mota, F.M.D., Auer, S., Breitman, K., Casanova, M.A.: Publishing Statistical Data on the Web. In: Proceedings, 6th IEEE International Conference on Semantic Computing. pp. 285–292. IEEE Computer Society (2012)
21. Tsymbal, A.: The problem of concept drift: definitions and related work. Tech. Rep. TCD-CS-2004-15, Computer Science Department, Trinity College Dublin (2004)
22. Wang, S., Schlobach, S., Klein, M.C.A.: What Is Concept Drift and How to Measure It? In: Knowledge Engineering and Management by the Masses - 17th International Conference, EKAW 2010. Proceedings. pp. 241–256. Lecutre Notes in Computer Science, 6317, Springer (2010)
23. Wilcoxon, F.: Individual comparisons by ranking methods. Biometrics Bulletin 1(6), 80–83 (1945)