

Analysing and Visualising Parliamentary Questions: A Linked Data Approach

Charlie Abela^(✉) and Joel Azzopardi

Department of Artificial Intelligence, University of Malta, Msida, Malta
{charlie.abela, joel.azzopardi}@um.edu.mt

Abstract. In many national parliaments, Members can exercise a basic Parliamentary function of holding the Executive to account by submitting Questions to Government Ministers. In certain parliaments, Members also have the faculty of either requesting a written answer or an oral one. Parliamentary Questions (PQs) often generate significant media attention and public interest, and are considered to be a very useful tool for parliamentarians to scrutinise the Government's operative and financial administration. Interesting insights about individual Members of Parliament (MPs) as well as about the Parliament as a collective institution can be gleaned by analysing PQs. In this paper we present a linked data approach to PQs that is complemented with visualisations intended to increase the accessibility, by citizens, to this rich repository of parliamentary data. We use PQ data from the Maltese Parliament ranging over the last four legislatures and present an application called *PQViz* that exploits graph analytics to expose interesting insights from this data.

Keywords: Linked data · Graph analytics · Visualisations
Parliamentary data

1 Introduction

Many national legislatures provide tools like Parliamentary Questions (PQs) intended to scrutinise the respective governments and keep them accountable [15]. Through PQs, individual Members of Parliament (MPs) table questions to the executive members of Government to scrutinise the Government's operative and financial administration. Over time, the number of questions tabled in parliament can become staggeringly high and keeping track of all the information that is found within these PQs becomes challenging. This is especially true for media in general that tend to pay considerable attention to the PQ sessions because the nature of PQs appear to generate increased level of citizens' engagement with the political process [16].

Although PQs are very central to the parliament's procedure, the content and nature of questions asked, vary considerably. It is in fact very common to find PQs related to issues specific to a particular locality that is part of the

asking MP's constituency, as well as PQs that are related to issues of national importance. This leaves the specifics of the questions asked and the reasons behind them open to conjecture [14].

Yet finding information about specific topics mentioned in PQs as well as who enquired about such topics and when, is no simple feat given that the number of PQs grows with every parliamentary sitting. In the last four years of the 12th legislature, parliamentarians from the Maltese Parliament made over 31K PQs in almost 500 parliamentary sessions. This data is made accessible through a web portal¹. Through the available interface, it is possible to apply filters over the PQ data, some of which are only useful if the user knows what to look for and/or is familiar with the whole PQ process. One such filter, filters PQs by sitting, which results in a list of parliamentary sittings that can then be expanded to display all the PQs that were asked during that sitting. Another tool available on this portal is a form-based search which requires that the user enters specific information about PQs and/or sittings, such as the PQ or sitting number, the date that a PQ was raised and who was the Minister to whom the PQ was asked. A keyword search is also available, however the result is a list of PQs that need to be expanded individually.

In this paper we report about research being conducted in the *apps4Parliament* project. This project is being done in collaboration with the Office of the Speaker (OS) within the Parliament of Malta who is the principal office holder of the House of Representatives. Through this research we intend to address some of the challenges mentioned in the previous paragraph, which are synonymous to archived data repositories. More specifically we want to provide a suite of tools to make parliamentary data more accessible and searchable for the common citizen. In this research we primarily focus on the PQ dataset and present the application entitled *PQViz*². In this application:

- i. PQ data is linked to MPs' profile data so that it is possible to more comprehensively display information about the MPs that asked questions and the Ministers that answered them;
- ii. graph analytics and visualisations are used to leverage on the interaction between MPs from the different parties and presents an interesting, interactive visualisation through which common citizens can more intuitively understand questions like: who asked whom about what, who asked the most PQs, and who answered them.

The rest of the paper is structured as follows. In Sect. 2 we present research which is closely related to our own and in Sect. 3 we provide information about PQs from the Maltese Parliament. This is followed with a detailed description in Sect. 4, of the linked data approach adopted to link PQ data. In Sect. 5 we describe how graph analytics and visualisations are used to make PQ data more easily accessible through the *PQViz* tool. Finally, in Sect. 6 we discuss how we intend to extend this tool to cater also for topic analysis over PQ data.

¹ <http://pq.gov.mt/>.

² https://parlament.opendatamalta.com/pqs_V2/.

2 Related Work

Government organisations publish different kinds of data to enable transparency and satisfy their legal obligations. Through portals like that of the UK government³, numerous datasets are being published to promote openness. This in turn has fostered the interest of the Web community which has over time become more and more intrigued with the idea of Open Government Data (OGD).

[1] states that it is important to open government data for three main reasons which include transparency, but also to release social and commercial value and to increase participatory governance. Initiatives like *Apps for Democracy*⁴ and *Show Us a Better Way*⁵ that promoted open data technologies have shown that OGD can be useful for different stake holders including citizens, businesses and government agencies.

There are however different challenges when opening up government data. Data content, format, structure and quality tend to vary considerably [2]. Exploring OGD is not trivial since it is imperative that the right sources are found to address the task at hand. The provision of OGD portals as reported by [1] is a step which many governments have undertaken. Such portals are intended to act as a one-stop shop to facilitate consumer's access to government data. The Global Open Data Index⁶ tracks whether published data is accessible to all stakeholders and measures a global level of openness. For the local Maltese sphere the Global Open Data Index of 2013, ranked Malta 21st with a score of 52%, however the National Data Strategy for Data-Driven Public Administration that deals with the opening up of Public Sector Information (PSI) is currently still being drafted⁷. One of the first challenges that we had to obtain Parliamentary data to reuse was precisely the lack of effective open data. The PQ data available on the Maltese Parliament's portal cannot be considered to be Open Data since it is not covered by an Open Data commons licence. It is possible to reproduce without charge or further permissions from the Maltese Government and/or the Maltese House of Representatives (MHoR) provided that the reproduced material is a true copy of the original data and that MHoR are identified as being the source. Other parliaments such as the UK Parliament provide an Open Parliament licence⁸ which allows for more flexibility in the way that parliamentary data can be used, reused and distributed.

Another important aspect highlighted by [5, 7] is the availability of datasets that are effectively linked along the Linked Open Data (LOD)⁹ principals defined by Tim Berners-Lee. The variety of data formats is however a bottleneck that needs to be addressed before interlinking the data. The most common non-proprietary formats include XML, CSV and JSON. For effective linking however,

³ <http://data.gov.uk>.

⁴ <http://www.appsfordemocracy.org/>.

⁵ <http://www.showusabetterway.co.uk/>.

⁶ <http://index.okfn.org>.

⁷ [https://mita.gov.mt/en/nationaldatastrategy/Documents/Data-DrivenPublicAdministration\(Malta\).pdf](https://mita.gov.mt/en/nationaldatastrategy/Documents/Data-DrivenPublicAdministration(Malta).pdf).

⁸ <http://www.parliament.uk/site-information/copyright/open-parliament-licence/>.

⁹ <https://www.w3.org/DesignIssues/LinkedData.html>.

an ontology based approach coupled with RDF serialisation is the most popular approach [2, 5, 7, 9]. Translation between these formats (and other related ones) and RDF was the focus of several research including [8, 17, 18].

Another aspect that has been found to be challenging for non-tech savvy web users is the lack of technical knowledge and understanding of the intricacies of LOD technologies. In research conducted by [4, 11] the main focus was on the need for suitable visualisation tools for Linked Data (LD). The former presents a comprehensive survey discussing efforts in the Semantic Web community to visualise LD and identifies how visual tool support can lead to more effective and intuitive user interaction with LD. The latter, on the other hand, presents visual tools developed as part of the LinDA project¹⁰, aimed at allowing SMEs to explore, visualise and analyse LOD for their daily tasks. In [13] the focus is on providing a tool called WebVOWL to visualise OWL ontologies using a force-directed graph layout. Through WebVOWL ontologies can be explored and optimised using various techniques.

OGD data related to the UK's general election was analysed in [10]. The authors used the RDF Open Cube vocabulary [3] to model multi-dimensional statistical data as RDF.

3 PQs in the Maltese Parliament

The first half hour of every parliamentary sitting of the Maltese Parliament are dedicated for questions. There are on average 65 MPs in the Maltese Parliament, however only MPs from the opposing benches and MPs from the Government's back-bench can pose questions to the cabinet¹¹. Every MP is entitled to put not more than six questions for oral answers for one sitting, provided that at least three days' notice is given. A MP may also submit questions for written answers.

Answered PQs are uploaded to the PQ portal via a system that is currently based on proprietary software. A document, with amongst other things, a copy of all the PQs answered during a particular sitting is also made available on the Parliaments' website¹².

The structure of every PQ consists primarily of a PQ ID, the date when the question was made, the sitting number, the title of the question, the name of the MP that asked the question and that of the Minister that answered, the question itself and the answer given. It is not uncommon for a PQ to also have attached to it, documents that the Minister presents to Parliament to comprehensively answer the question. Listing 1.1 is an XML representation of a particular PQ with number 27059, that was asked on 29th July 2016 during sitting number 446 of the current (XII) legislature. Note that both question and answer in this example are being reproduced in English, nevertheless, Maltese PQs are saved in the native language.

¹⁰ <http://linda-project.eu/>.

¹¹ Cabinet includes the Prim Minister and Ministers.

¹² A copy of the PQs asked during the sitting held on 8th March 2017 is found here: <http://parlament.mt/sittingdetails?sid=5764&l=1&legcat=13&forcat=12>.

```

1 <pq>
2 <id>C1257D2E0046DFA1C125805E00275E7E</id>
3 <pq_written_reply>N</pq_written_reply>
4 <reply_category_id>FINAL</reply_category_id>
5 <pq_no>27059</pq_no>
6 <pq_sitting_date>02/11/2016 09:00</pq_sitting_date>
7 <pq_sitting>446</pq_sitting>
8 <pq_raised_by_gender>M</pq_raised_by_gender>
9 <pq_raised_by_name>ANTHONY</pq_raised_by_name>
10 <pq_raised_by_surname>AGIUS DECELIS</pq_raised_by_surname>
11 <pq_min_name>Ministry for Economy, Investment and Small Enterprises</pq_min_name>
12 <pq_minister_name>CHRIS CARDONA</pq_minister_name>
13 <pq_category_id>ORAL</pq_category_id>
14 <pq_date_raised>29/07/2016</pq_date_raised>
15 <pq_raisedby_mp_id>AGDA</pq_raisedby_mp_id>
16 <pq_addressed_to_ministry_id>EIIZ</pq_addressed_to_ministry_id>
17 <pq_heading>Commerce Licences</pq_heading>
18 <pq_legis_no>XII</pq_legis_no>
19 <pq_text>Can the Minister say how many commercial licences were issued from the year 2013
    till this date?</pq_text>
20 <reply_text> I inform the Honory Gentleman that I am placing this information on the Table
    of the House.</reply_text>
21 </pq>

```

Listing 1.1: PQ structured data

4 PQViz: Linking Parliamentary Questions

A PQ by itself is as important as the content that it includes, and even then, it is not easy to appreciate how important is the topic, who are the players involved (who's asking and who is answering), how many other PQs deal with the same topic, whether other MPs have asked questions about this same topic to this Minister or to other Ministers. For this reason tools need to be made available so that this information can be extracted.

As pointed out in Sect. 1, the existing PQ portal does provide tools through which this information can be extracted, yet they are not intended for the common citizen and are time consuming to use. One way that information about PQs can be given more context and made more comprehensive is by adopting a linked data approach [1]. This is the approach that we have adopted in the *PQViz* application. The architecture of the application is depicted in Fig. 1 and in the following sections we will discuss the challenges that we had to address, and the solutions that we adopted to transform and link PQ data.

4.1 Harvesting Parliamentary Data

Through the collaboration with the OS we have been given access to the PQ web service. A batch process is executed on the server side at weekly intervals so that the system is always updated with the latest list of PQs. The format of the PQ data is XML and is similar to the excerpt shown in Listing 1.1.

The XML of new PQs is preprocessed and cleaned to make sure that any formatting issues are resolved. The main issues that needed to be tackled included the following:

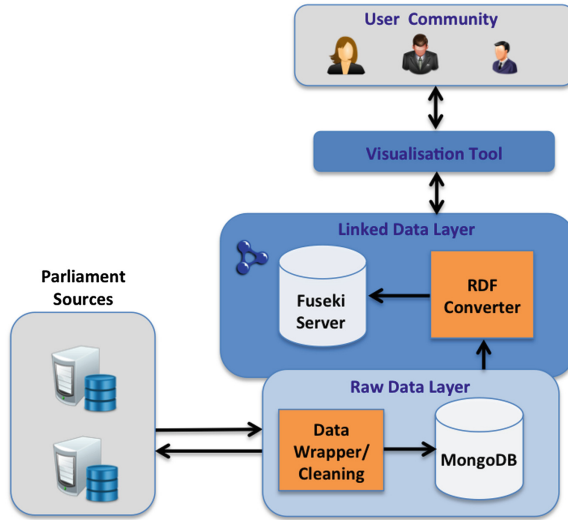


Fig. 1. PQViz architecture

- Whilst the PQ data provides an MP ID uniquely made up of the name and surname of the MP who raised the question, it did not provide a similar ID for the minister who answered the question. Only the minister's name is provided and although a ministry ID is provided, it is not uncommon that this changes when there is a change in the portfolio name, for instance after there is a cabinet reshuffle;
- There was a degree of inconsistency on the usage of MP IDs, names and ministry descriptions used. Apparently the initial data entry interface used by parliamentary staff did not enforce the use of certain specified vocabularies when inserting such data. This resulted in multiple variations of the same name/description, for example: Angelo Farrugia and Anġlu Farrugia. There were also cases of errors in the IDs used which resulted in multiple IDs being associated to the same MP.
- No details were provided on the party membership of the MPs submitting the PQs and the ministers answering the PQs.

To address these issues, and also to simplify the querying mechanism, a separate collection was created to hold the details of every MP (including cabinet members) within each legislature. The profile data for each MP was extracted using dedicated wrappers from the relevant pages on Parliament's web portal. The pages that were scraped were those that included profile details for each MP in the current legislature, as well as pages that included details of MPs that formed part of past legislatures. The collected details included also party details, professional titles, email addresses, personal websites (when available) and contact phone numbers. Of particular interest was the information about the roles that MPs occupied, that is whether they were Ministers or Junior Ministers, when they were appointed and when they ceased to occupy those roles.

Each MP was then assigned a single unique ID that was taken from the PQ collection when available and/or otherwise generated using the templates used from the provided MP IDs. These IDs were then checked for uniqueness.

Both the MP and PQ data collections are stored in a MongoDB¹³ database. We have adopted this approach primarily to have a backup of the clean data, but also to have the possibility to try out different options. In MongoDB data stored as a JSON objects and converting between XML and JSON is a straight forward process that is handled by a simple transformation class that we have implemented. Once the MPs collections are generated, the PQs collection was cleaned from errors and inconsistencies and harmonised with the details in the MPs collections. This process was performed by submitting relevant “Aggregate” queries to MongoDB on the PQs collection, and then updating the MongoDB collection to remove inconsistencies and erroneous variations in names and IDs.

4.2 Linked Data Model

The RDF Converter component shown in Fig. 1 automatically extracts the MP objects and the latest set of PQs from the MongoDB database and transforms every JSON object to RDF.

MP profile data does not usually change that often, thus this transformation process is only performed whenever some parliamentary reshuffle occurs. During the 12th legislature for example, this happened a number of times, whereby a Minister’s portfolio was changed, or a Junior Minister was demoted and/or promoted.

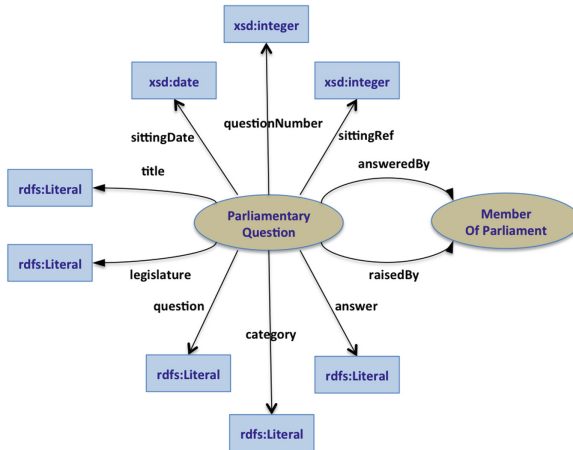


Fig. 2. PQ model

¹³ <https://www.mongodb.com/>.

```

1 <rdf:Description rdf:about="&mp;mp00000">
2   <rdf:type rdf:resource="&dbo;MemberOfParliament"/>
3   <foaf:name>Jesmond Mugliett</foaf:name>
4   <foaf:title>Mr</foaf:title>
5   <foaf:firstName>Jesmond</foaf:firstName>
6   <foaf:surname>Mugliett</foaf:surname>
7   <mp:party rdf:resource="&dbr;Nationalist_Party_(Malta)"/>
8   <mp:elected>
9     <rdf:Description>
10       <mp:legislature>VIII</mp:legislature>
11       <mp:electedDate>29-10-96</mp:electedDate>
12       <mp:oathofAllegiance>05-12-96</mp:oathofAllegiance>
13       <mp:dissolution>03-08-98</mp:dissolution>
14     </rdf:Description>
15   </mp:elected>
16   <mp:elected>
17     <rdf:Description>
18       <mp:legislature>IX</mp:legislature>
19       <mp:electedDate>08-09-1998</mp:electedDate>
20       <mp:oathofAllegiance>24-10-1998</mp:oathofAllegiance>
21       <mp:dissolution>15-04-2003</mp:dissolution>
22       <mp:ministerRole>
23         <rdf:Description>
24           <mp:appointedRole>24-09-1998</mp:appointedRole>
25           <mp:roleTitle>Parliamentary Secretary in the Ministry of Education</mp:roleTitle>
26         </rdf:Description>
27       </mp:ministerRole>
28     </rdf:Description>
29   </mp:elected>
30   <mp:elected>
31     <rdf:Description>
32       <mp:legislature>X</mp:legislature>
33       <mp:electedDate>14-04-2003</mp:electedDate>
34       <mp:oathofAllegiance>24-05-2003</mp:oathofAllegiance>
35       <mp:dissolution>09-03-2008</mp:dissolution>
36       <mp:ministerRole>
37         <rdf:Description>
38           <mp:appointed>15-04-2003</mp:appointed>
39           <mp:roleTitle>Parliamentary Secretary in the Ministry of
40             Education</mp:roleTitle>
41         </rdf:Description>
42       </mp:ministerRole>
43       <mp:ministerRole>
44         <rdf:Description>
45           <mp:appointedRole>20-03-2004</mp:appointedRole>
46           <mp:roleTitle>Minister for Youth and the Arts</mp:roleTitle>
47         </rdf:Description>
48       </mp:ministerRole>
49     </mp:elected>
50   <mp:elected>
51     <rdf:Description>
52       <mp:legislature>XI</mp:legislature>
53       <mp:electedDate>12-03-2008</mp:electedDate>
54       <mp:oathofAllegiance>10-05-2008</mp:oathofAllegiance>
55       <mp:dissolution>07-01-2013</mp:dissolution>
56     </rdf:Description>
57   </mp:elected>
58 </rdf:Description>

```

Listing 1.2: MP RDF profile

To perform the translation from JSON to RDF we considered different research solutions [2, 12, 17, 18]. We decided to use the RDF Translator¹⁴ discussed in [17] that is based on a Python library through which it is possible to translate between different serialisations including JSON and RDF/XML.

¹⁴ <http://rdf-translator.appspot.com/>.

The model depicted in Fig. 2 represents the PQ model used to generate an RDF version of a PQ. Every PQ is associated with the *ParliamentaryQuestion* type and assigned a unique ID based on the legislature and the PQ number assigned to it by Parliament. The PQ number is reset with every new legislature. A PQ is *raisedBy* and *answeredBy* by a *dbo:MemberOfParliament* (refer to the MP profile model discussed below). Other properties of interest include the actual *question* and *answer* which are currently considered to be Literals. Nevertheless, it is not uncommon for PQ answers to include information in a tabular format as well as attached documentation. We intend to cater for such data in a future release of the application.

As stated in Sect. 4.1, the MP profile is composed of data extracted from different parliamentary sources. The MP profile Listing 1.2 refers to a MP that belongs to the Nationalist party, linked to *dbr:Nationalist_Party_(Malta)* and has been elected in four legislatures. For every legislature the profile includes information about the date that the MP was elected, the date that he/she took the oath and the date when his term ended. The profile also includes information about the roles, apart from that of an MP, that the MP occupied during each legislature. In some cases such as during the 10th legislature, the MP in question occupied two different roles, that of a Parliamentary Secretary (i.e. a Junior Minister) and that of a Minister. In the near future we plan to introduce other personal information about every MP such as email, phone number and personal website.

The RDFied, PQ and MP data is then persisted in an Apache Fuseki server¹⁵ and is queried by the visualisation layer through SPARQL.

5 PQViz: Interaction Graph

PQViz exploits PQ and MP data to construct a graph that represents the interactions between the MPs. Every node in the graph represents an MP and the edge represents the interaction, based on PQs asked and answered.

The graph structure is complimented with a visualisation layer that is currently based on D3¹⁶. This popular Javascript library accepts a JSON representation of a graph and renders the structure using a variety of possible layouts.

In *PQViz* we use bubbles and a force-directed graph layout to make the graph interactive. Currently we are considering PQs from the 9th up till the 12th legislature¹⁷. The user can immediately view the topmost 15 MPs that have either asked or answered the most PQs during a particular legislature (Fig. 3). The force-directed layout can however be used to drill down to specific MP interactions to visualise which MPs asked questions to whom as can be seen in Fig. 4.

The application uses a number of visualisation cues to display the information. The size of the bubble depends on a computed value that takes into consideration

¹⁵ <https://jena.apache.org/>.

¹⁶ <https://d3js.org/>.

¹⁷ We will soon be considering the 13th legislature which has just started.

the number of questions asked and/or answered, while the colour of the bubble is the same as that of the MP's party. When the user clicks on the interaction between two specific MPs, the application displays the profiles of both MPs and the actual PQs that were asked. The user can scroll through this list and go through the details of each PQ.

From the interaction graph, it is possible to extract a number of statistics that are currently not being displayed, but which can be of interest to the users. Apart from the total number of questions asked and/or answered over each legislature by individual MPs, it is possible to compute the number of questions asked by (and answered by) one particular MP to another. This information, complimented with information about the topics discussed in the PQs (we refer



Fig. 3. Topmost MPs that asked the most PQs

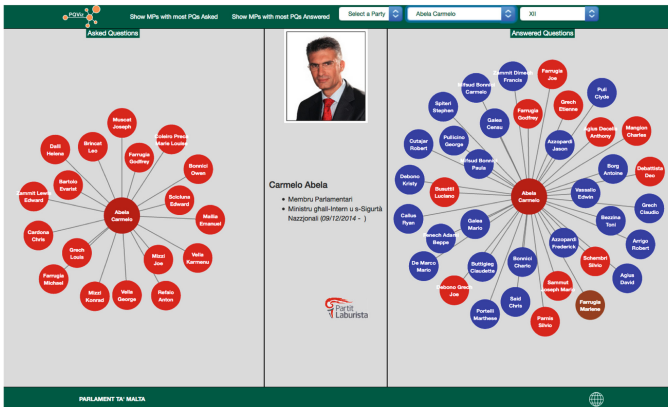


Fig. 4. Interaction graphs for a particular MP

the reader to Sect. 6), is useful to understand better the individual MPs' interests and how these interests potentially changed over time.

6 Future Work

PQViz exists both as a web-based application, however it can be downloaded as a mobile app from both Google's Play Store and Apple Store. We are furthermore continuing our efforts to extend *PQViz* with a set of new features to make it more useful and interesting. One such feature will take into consideration the topics of the PQs. This is somewhat of a challenge because PQs are in Maltese and unfortunately there aren't many tools available to work with the language. We intend to leverage on existing tools found on the Maltese Language Resources server (MLRS)¹⁸ which is an ongoing effort intended to increase the number of natural language processing tools for Maltese. We intend to introduce a search facility to allow for these topics to be found and interesting results to be generated. We would like to provide answers to questions such as which MP asked the most about specific topics, the time windows when such topics were considered to be hot topics and the possibility to identify communities of MPs that have dealt with similar topics over time. The idea is to use community detection algorithms similar to those discussed in [6]. We also intend to exploit relevant and related information about specific topics found in other data sources which include other parliamentary documents and debates, as well as articles from news portals.

PQViz is the first of a number of apps being developed as part of the *apps4Parliament* project. These apps are intended to make parliamentary data more accessible and to increase the knowledge of the common citizens about parliamentary governance.

Acknowledgments. This work is part of the *apps4Parliament* project and is being done in collaboration with the Office of the Speaker (Parliament of Malta) and co-funded by the Malta Information Technology Agency (MITA).

References

1. Attard, J., Orlandi, F., Scerri, S., Auer, S.: A systematic review of open government data initiatives. *Gov. Inf. Q.* **32**(4), 399–418 (2015)
2. Böhm, C., Freitag, M., Heise, A., Lehmann, C., Mascher, A., Naumann, F., Ercegovac, V., Hernández, M.A., Haase, P., Schmidt, M.: GovWILD: integrating open government data for transparency. In: *WWW (Companion Volume)*, pp. 321–324. ACM (2012)
3. Cyganiak, R., Reynold, D.: The RDF Data Cube Vocabulary. W3C Working Draft (2014). <http://www.w3.org/TR/vocab-data-cube/>
4. Dadzie, A., Rowe, M.: Approaches to visualising linked data: a survey. *Semant. Web* **2**(2), 89–124 (2011)

¹⁸ <http://mlrs.research.um.edu.mt/>.

5. Ding, L., DiFranzo, D., Graves, A., Michaelis, J., Li, X., McGuinness, D., Hendler, J.: Data-gov Wiki: Towards linking government data. In: Proceedings of AAAI Spring Symposium on Linked Data Meets Artificial Intelligence (2010)
6. Fortunato, S.: Community detection in graphs. *Phys. Rep.* **486**(3–5), 75–174 (2010). ISSN 0370–1573
7. Hall, W., Schraefel, M.C., Gibbins, N., Berners-Lee, T., Glaser, H., Shadbolt, N., O’Hara, K.: Linked open government data: lessons from Data.gov.uk. *IEEE Intell. Syst.* **27**(3), 16–24 (2012)
8. Hasapis, P., Fotopoulou, E., Zafeiropoulos, A., Mouzakis, S., Koussouris, S., Petychakis, M., Kapourani, B., Zanetti, N., Molinari, F., Virtuoso, S., Rubattino, C.: Business value creation from linked data analytics: The LinDA approach. In: eChallenges e-2014 Conference, Belfast, Northern Ireland, 29–31 October 2014
9. Hoxha, J., Brahaj, A.: Open government data on the web: a semantic approach. In: Proceedings of the 2011 International Conference on Emerging Intelligent Data and Web Technologies (EIDWT 2011), pp. 107–113. IEEE Computer Society, Washington, DC (2011)
10. Kalampokis, E., Tambouris, E., Tarabanis, K.: Linked open government data analytics. In: Wimmer, M.A., Janssen, M., Scholl, H.J. (eds.) EGOV 2013. LNCS, vol. 8074, pp. 99–110. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40358-3_9
11. Thellmann, K., Orlandi, F., Auer, S.: LinDA - visualising and exploring linked data. In: Proceedings of the Posters and Demos Track of 10th International Conference on Semantic Systems - SEMANTiCS 2014. CEUR-WS, vol. 1224, pp. 39–42 (2014)
12. Lanthaler, M., Gütl, C.: On using JSON-LD to create evolvable restful services. In: Proceedings of the 3rd International Workshop RESTful Design, pp. 25–32 (2012)
13. Lohmann, S., Link, V., Marbach, E., Negru, S.: WebVOWL: Web-based visualization of ontologies. In: Lambrix, P., Hyvönen, E., Blomqvist, E., Presutti, V., Qi, G., Sattler, U., Ding, Y., Ghidini, C. (eds.) EKAW 2014. LNCS (LNAI), vol. 8982, pp. 154–158. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-17966-7_21
14. Martin, S.: Parliamentary questions, the behaviour of legislators, and the function of legislatures: an introduction. *J. Legislative Stud.* **17**(3), 259–270 (2011)
15. Norton, P.: Introduction: Parliament since 1960. In: Franklin, M., Norton, P. (eds.) *Parliamentary Questions*. Clarendon Press, Oxford (1993)
16. Salmond, R.: *Parliamentary Question Times: How Legislative Accountability Mechanisms Affect Mass Political Engagement*. Unpublished paper, University of Michigan (2010)
17. Stolz, A., Rodriguez-Castro, B., Hepp, M.: RDF Translator: A RESTful Multi-Format Data Converter for the Semantic Web. Technical report, E-Business and Web Science Research Group (2013)
18. Weaver, J., Tarjan, P.: Facebook linked data via the graph API. *Semant. Web* **4**(3), 245–250 (2013)