Lessons Learned from a Knowledge-driven Search Application on-top of Large Data Sets

Dennis Diefenbach, Pierre Tardiveau, Andreas Both, Kamal Singh, Pierre Maret

Lessons Learned from a Knowledge-driven Search Application on-top of Large Data Sets

DENNIS DIEFENBACH, Université de Lyon, CNRS UMR 5516 Laboratoire Hubert Curien, France
PIERRE TARDIVEAU, Telecome Saint-Étienne, France
ANDREAS BOTH, DATEV eG, Germany
KAMAL SINGH, Université de Lyon, CNRS UMR 5516 Laboratoire Hubert Curien, France
PIERRE MARET, Université de Lyon, CNRS UMR 5516 Laboratoire Hubert Curien, France

The Web stores huge amounts of data. Additionally the number of stored data is increasing permanently. Hence, while using a search application, after some time it is likely that users are confronted with unknown instances or properties while triggering their search questions. From this observation the research challenge is derived, how data can automatically be visualized for a user without leaving the search application. Here, we will present observations from a search-driven application from the field of Question Answering on-top of the Web of Data. This Web application is using Wikidata – a data source derived from Wikipedia – and several other to provide access to general knowledge and specific knowledge from particular domains. Hence, the size of the data set is very large. The problem is how to tackle the sheer amount of available instances and properties (volume), the high variety due to the ambiguity of natural language questions, and the broad field represented by a general-purpose knowledge base.
Data (instances and properties) need to be visualized so that it can be explored with respect to different dimensions and allowing different granularity. Additionally, feedback interaction points were required to make the system learn over time and deal with the ambiguity of natural language questions. Concluding, in this paper we will provide an overview of the challenges we have identified and the derived solutions.

CCS Concepts: • **Information systems** → *Web searching and information discovery*; • **Human-centered computing** → *Human computer interaction (HCI)*; *Visualization systems and tools*;

Additional Key Words and Phrases: Question Answering, User Interface, Big Data, User Feedback

1 INTRODUCTION

In the past user interfaces (UI) were providing the success to a predictable data set. UI designers and developers were aware of the variety and volume of the data as well as the velocity of the data. This is fundamentally changing due to the characteristics of Big Data and Big Data applications. Corresponding challenges were previously discussed, e.g., in [3, 5, 23]. We primarily use the common classification of the "3 Vs of Big Data" challenges[1] (cf., [17]) as used by many other authors ([17, 18, 22]) to highlight some of the challenges from the visual perspective:

[1]It is well-known that several other classifications are available, e.g., [23] uses 4V and [7, 21] use 5V.

Authors' addresses: Dennis Diefenbach, Université de Lyon, CNRS UMR 5516 Laboratoire Hubert Curien, Saint-Étienne, France, dennis.diefenbach@univ-st-etienne.fr; Pierre Tardiveau, Telecome Saint-Étienne, Saint-Étienne, France, pierre.tardiveau@telecom-st-etienne.fr; Andreas Both, DATEV eG, Nuremberg, Germany, andreas.both@datev.de; Kamal Singh, Université de Lyon, CNRS UMR 5516 Laboratoire Hubert Curien, Saint-Étienne, France, kamal.singh@univ-st-etienne.fr; Pierre Maret, Université de Lyon, CNRS UMR 5516 Laboratoire Hubert Curien, Saint-Étienne, France, pierre.maret@univ-st-etienne.fr.

XXXX-XXXX/2018/11-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

- The volume of the data might be beyond the scale a human will ever be capable to process (in this paper referred as Challenge 1). For example, the number of entities in Wikipedia/Wikidata is around 47 Million[2]. Hence, a singular person cannot know all of them nor is it possible to show even fractions of this set in a search result.
- The variety of the data is expanding in beyond a limit which can be presented in a reasonable way (Challenge 2). For example, the Wikipedia entity "Florence" (Italian city, located in Tuscany) contains 530 properties[3] (e.g., the continent, head of government and the time zone). Hence, even visualizing information about a single entity need to be addressed to enable a proper user understanding and interaction.
- The velocity of the data is increasing (Challenge 3). For example, data integration process are adding new instances to the considered data set due to the permanently growing of data in general[4].
- Consequently the processes evaluating the data are very complicated and not easy-to-understand for common users. Hence, the required trust needs to be addressed by interface components providing insights of the data leading to establish confidence in the provided application. This challenge is not covered by the 3V classification. In this paper, we refer to it as Challenge 4.

In this paper we will provide good practices for applications on-top of Big Data. These insights are derived from a Question Answering application using the data of the Wikipedia/Wikidata Knowledge Base (KB) to answer factoid questions. Hence, it covers general knowledge considered to be relevant for mankind (Wikidata: zipped file size ca. 23 GB[5]). A concrete example would be to answer a question like *"Give me museums in Florence?"* using information contained in a Knowledge Base like Wikidata. As the used data represents actual facts, i.e., knowledge, we refer to our approach as knowledge-driven.

We elaborate our insights while using the prototypical UI called *Trill*[9], a user interface that can be used to display the results of QA systems over large KBs. All screenshots in this publication show Trill combined in the backend with the QA system WDAqua-core1[10]. A live demo is available under:

www.wdaqua.eu/qa

The paper is organized as follows. In Section 2 we give a brief overview of related work. The Section 3 and 4 address the challenges mentioned above. The first provides insights to the visualizations while the second addresses the interaction points built into the UI. Examples of Trill are provided each time to show an example. Conclusions and future work are provided in Section 5.

2 RELATED WORK

In Big Data environments visualizations are important to cope with the volume, variety and velocity of the data. For example [1] present visualization techniques to explore the human connectome, i.e., how the neurons in the human brain are connected. [19] present visualization techniques for scientists to better understand aspects of tropical meteorology from large-scale spatiotemporal climate simulation data. Finally [16] describes methods for the visualization of fluid dynamics.

Just a few works offer an enhanced UI although the challenges beyond traditional search are well-known (e.g., [27]) and addressed by UIs with enhanced visual components (e.g., [15, 21]).

In the domain of QA users also have to deal with huge amounts of data. In the last years a big effort was made to develop high quality QA systems capable of answering natural language questions using unstructured or (semi-)structured data. Also in the more particular domain of QA over KBs this is the case. One can count more then 40 systems evaluated on public available benchmarks. For an exhaustive list we refer to [12]. Most of these

[2]Number from the 28/04/2018. Updated statistics can be found at www.wikidata.org

[3]query.wikidata.org/#SELECT%20%28count%28%2a%29%20as%20%3Fc%29%0AWHERE %20%7B%0A%20%20wd%3AQ2044%20%3Fp%20%3Fo%20.%0A%7D

[4]cf., tools.wmflabs.org/wikidata-todo/stats.php

[5]cf., wikidata.org/wiki/Wikidata:Database_download

works focus on the translation of natural language question to formal representations, particularly to SPARQL queries. Some exceptions that we are aware of are *SINA*[24], QAKiS[4] and *Platypus*[6]. In the following we use the user interface (UI) named *Trill* for providing examples. It might be considered as reference implementation of how to cope with the large amount of data that a user could have to deal with when large answer sets are computed. While the UI of the above cited QA systems only provide simple representations for the answers, Trill is able to present answer-data sets from different dimensions and with different granularities.

3 VISUALIZATION

An important aspect in QA over KB is the requirement to present the answers of a possible complex data structure to a (regular) user. Note that most research focus on the process of answer generation, i.e., to derive from a natural language question a formal representation (e.g., a SQL query for Relational Database Management Systems, or SPARQL for triplestores). If this is the first step to make data accessible to end users, it is certainly not the last. For example, the answer of a SPARQL query is a set of URIs which are the identifiers of the actual data of the corresponding result item, e.g., the unique identifier of the city of Florence (Italy) is https://www.wikidata.org/wiki/Q2044. Via this URI a large numbers of information can be retrieved, e.g., for Florence there are 530 associated properties. Clearly only some of these can be shown to the user, because the volume and variety of the data is too large (cf., Challenge 1).

For designing the UI it is required to answer the following question: Which information should be presented to the user, s.t., it provides an acceptable user experience.

3.1 Aggregation of Answer Sets

In the case of multiple entities we offer three different ways to visualize the answer set. We call these different ways answer aggregations.

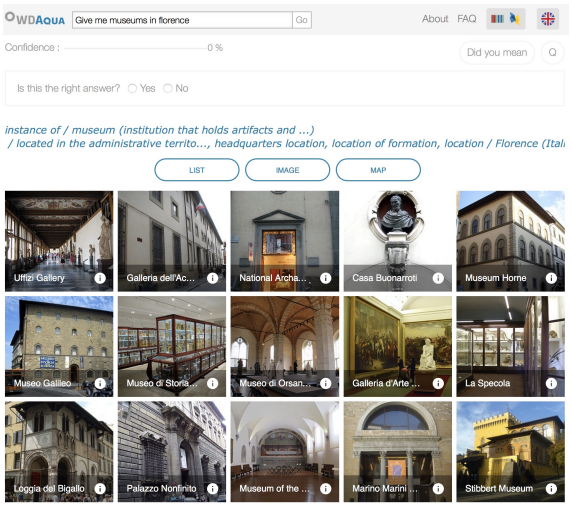
(1) The first aggregates the labels, and consist of the list of labels of the answer items. An example is shown in Figure 1c.
(2) The second aggregates the pictures of the answer entities. The answer entity depictions are shown near each other with a small textual label. An example is shown in Figure 1a.
(3) Finally the third aggregation is happening on the geographic coordinate level. We aggregate the geographic coordinates and we display the points together in a map. By clicking on the points the name of the entity appears. An example is shown in Figure 1b.

Clearly, this approach addresses the Challenge 1. From (1) – default solutions for representing data – to (2) and (3).

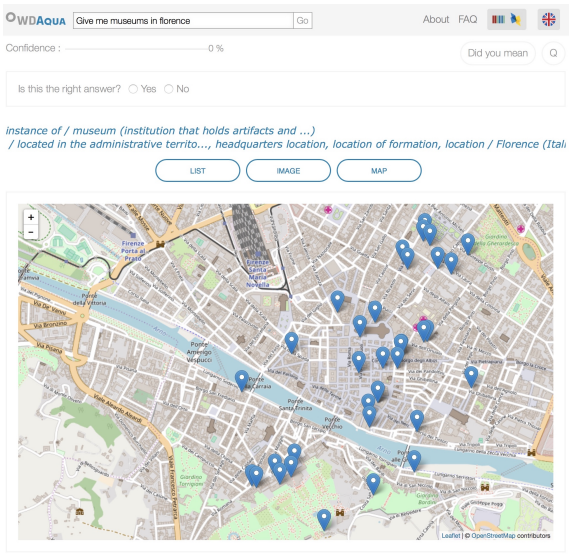
The aggregation level of images (2) is higher in contrast to textual representations. Additionally visual representations of a search result item are decreasing the cognitive interpretation time in contrast to textual representation and the images out of the search context are expected to be identified faster (cf., [2, 6, 25]). Hence, despite the larger number of images within the viewport of the user it can also be assumed that users will process the search result visualized by images faster. The aggregation level of a map is even higher. If required, a map of the whole world can be presented visualizing huge numbers of items (e.g., following [20]).

However, it is not always meaningful to show a particular type of aggregation. While the labels of the items within the answer set can always be shown, this is not the case for images and geographic coordinates. For example, for a list of persons the aggregation for geographic coordinates is meaningless although geospatial properties are attached to a person entity (e.g., location of birth or death, various travel locations). To decide which aggregation to show the ratio of entities having associated image and geographic coordinates are computed.

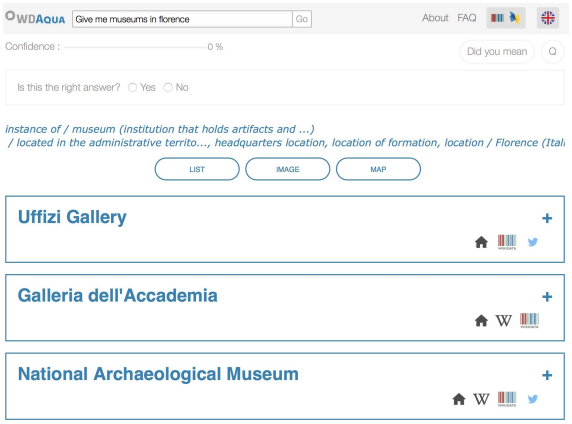
[6]http://sina.aksw.org, http://live.ailao.eu, http://qakis.org/qakis2/, https://askplatyp.us/?



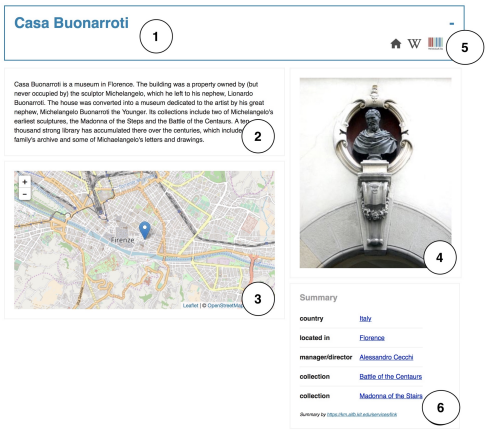
a) UI element showing the museums in Florence aggregated by image. The user can see for each museum a picture, the name of the museum and, by clicking con the "i" icon, get more information about the entity.



b) UI element showing the museums in Florence aggregated by geo-location. The user can see how the museums are distributed in space.



c) UI element showing the museums in Florence aggregated by label. To get more information about a museum it suffices to click on the corresponding instance.



d) This figure shows all the information that are shown to the user for the answer entity “Casa Buonarroti”, a museum in Florence.

Fig. 1. The figures show ways representing large data sets using the aggregation methodology for multiple entities – a), b), and c) – using the question “Give me museums in Florence”, while d) is showing the representation of one item of the result set.

If the ratio is above a predefined threshold, the corresponding aggregation is displayed. The aggregation are prioritized as follows:

(1) If the number of available item depictions is beyond the predefined threshold, the depictions are shown.
(2) Else if the number of available geographic coordinates is beyond the predefined threshold, a map showing the search result items as pins is presented to the user.
(3) Else the list of labels of the items of the result set is shown.

The presented aggregations improve the capabilities to handle Challenge 1, the volume of the data, and thanks to the adaptation to the type Challenge 2, the variety of the data, is addressed too. Note: By aggregating the data on high-level concepts and by the adaptation of the information shown depending on the semantic types associated, also Challenge 3, the velocity, can be handled.

3.2 Answer Set Item

Each of the aggregations allow to explore more in depth a specific entity. We explored existing KBs like Wikidata and DBpedia and found that there is a number of useful facts that can be directly shown to the user without being overwhelming. The following information (if available) are shown (cf., Figure 1d):

(1) the label of the answer entity in the corresponding language,
(2) a textual description retrieved from Wikipedia,
(3) a map indicating the location of the entity taken from OpenStreetMap,
(4) an image depicting the entity, this represents the best way to immediately identify the answer entity.
(5) external links related to the entity are shown like the webpage, the wikipedia entry, to Twitter, to Facebook, to Instagram, to Google Scholar and ORCID.

Finally for example for songs (like Longview (song by Green Day)) a video is shown.[7] These information not only add context to the answer entity increasing the confidence of a user, but also give direct opportunities to consume this information which are directly linked to the searched entity.

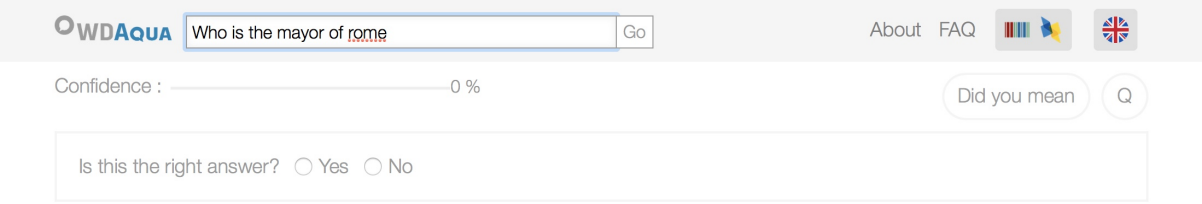
However, despite the properties manually picked due to the indisputable importance, there might be a huge number of additional properties (cf., Challenge 2, Section 1). For this reason a mechanism is required to distinguish more important properties from less important. Consequently we introduces an approach to compute the top-k most relevant facts for the entity and only present them to the user. These properties are computed using the SummaServer [14]. This does not only provide some relevant facts to contextualize the entity but also increase discoverability of the dataset itself. The user can browse through the propose facts and discover the dataset in a serendipitous way. In Trill k was set to 5 as shown in Figure 1d.

Hence, the answer presentation is also adapting to the properties of data addressing Challenge 1 and Challenge 2.

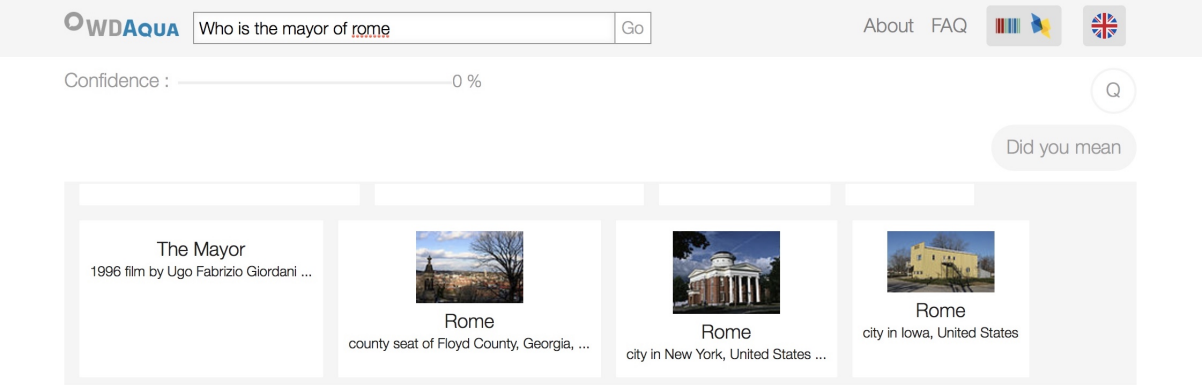
4 USER INTERACTION AND FEEDBACK

In this section we are describing the interaction possibility that we implemented in Trill [11]. The goal of these interaction methods is to handle the huge amount of information which we have to deal with. Note, in applications built upon Big Data mostly collecting feedback is crucial due to underlying AI algorithms (here the Question Answering) and its improvement.[8] Hence, each user feedback is added to the training data to be used for improving the search quality (cf., [26]) after the next release of the UI.[9] This follows the high-level intention to increase the data accessibility by improving the system's quality (over time) w.r.t. computed answer set while

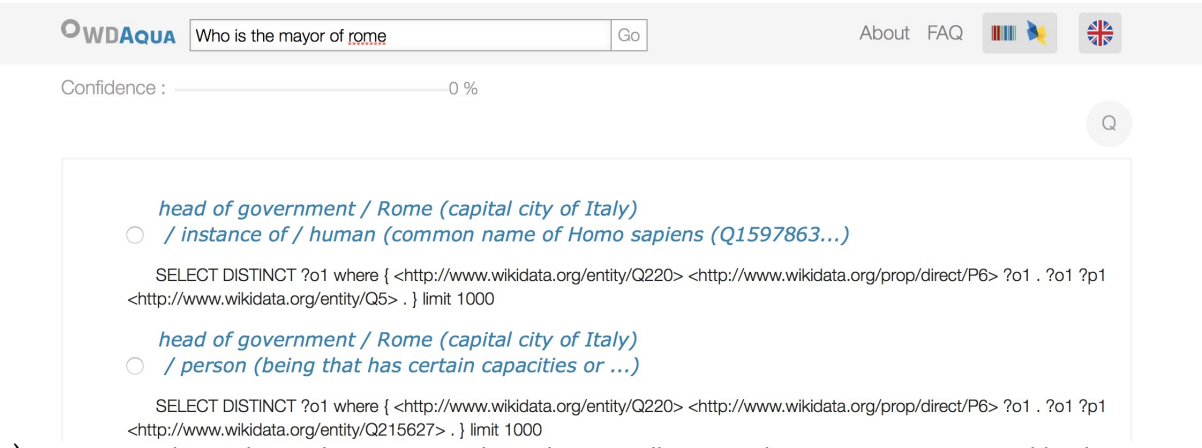
[7] Longview (song by Green Day)
[8] In Trill the collected log files are used to construct a new benchmark called *WDAqua-core0* that is described in [13] and is available under [8].
[9] Note, even "wrong" feedback of users will not crash this approach as machine learning algorithms are capable of handling conflicting data.



a) Whenever a user asks a question, a simple box is appearing asking feedback to the user. It can be seen in the lower part of this figure.



b) The "Did you mean" functionality allows a user to interact with the QA system to specify specific entities the question refereed to. The interpretation is then adjusted.



c) Expert users that understand SPARQL can choose between all generated SPARQL query generated by the question answering system. By clicking on the corresponding SPARQL query the answer set is recomputed.

Fig. 2. Different types of user feedback interfaces that are integrated into Trill. a) shows a very simple user feedback interface that is used to collect training data. b), c) present interfaces that can be used when the question is ambiguous.

exploiting the user interaction. Hence, implementing touchpoints for users needs to be addressed. Here, three methods for feedback collection are presented.

The first interaction method is addressing common user (i). Therefore, the requirements for interaction are kept intentionally very low. When a user asks a question and the answer is computed, a easy-to-understand form is shown presenting the user the option to provide feedback whether the computed answer is correct or wrong. An example is given in Figure 2a.

The second interaction is used to engage the user when the question was not interpreted correctly by the QA system. To deal with them we offer two interfaces. One addresses experienced end users and the other one expert users (i.e., users that are able to understand the technical query language used in the backend).

Experienced end users can interact with the QA system to specify specific entities the question referred to, i.e., they are clarifying the context/interpretation of the current question (ii). Note that this problem particularly arises on KBs containing large volumes of data since the user's question can be interpreted in many different ways (e.g., the question *"How old is Paris?"* might be interpreted as the request for the age of a place, a person, a mythical hero, etc.). The interpretation is then adjusted. An example is given in Figure 2b.

In the expert interface (iii) multiple queries generated by the QA systems are listed (in Trill: SPARQL queries are shown). By clicking on them an expert user triggers the computation and visualization of the corresponding answer set. This way an expert user can interact with the system and find an answer even if the system could not directly provide one. An example is given in Figure 2c.

While providing these user touchpoints – (i), (ii), and (iii) – we are addressing Challenge 4.

5 CONCLUSION

In this paper we gave an overview about typical challenges which need to be addressed while implementing applications driven by Big Data. To provide insights to the good practices experienced, we used a real-world question answering system to address the four identified challenges while establishing applications on-top of Big Data. The system covered the collected requirements w.r.t. (i) the presentable number of data elements in a result set (aggregation), (ii) the variety of the property of data elements (selection), and (iii) different depth of involvement derived from the capabilities of the expected user groups (regular user, experienced user, expert user).

Acknowledgments Parts of this work received funding from the European Union's Horizon 2020 research and innovation programme under the Marie SkÅĆodowska-Curie grant agreement No. 642795, project: Answering Questions using Web Data (WDAqua).

REFERENCES

[1] Johanna Beyer, Markus Hadwiger, Ali Al-Awami, Won-Ki Jeong, Narayanan Kasthuri, Jeff W Lichtman, and Hanspeter Pfister. 2013. Exploring the connectome: Petascale volume visualization of microscopy data streams. *IEEE computer graphics and applications* 33, 4 (2013), 50–61.

[2] David Beymer, Peter Z. Orton, and Daniel M. Russell. 2007. An Eye Tracking Study of How Pictures Influence Online Reading. In *Human-Computer Interaction – INTERACT 2007*, Cécilia Baranauskas, Philippe Palanque, Julio Abascal, and Simone Diniz Junqueira Barbosa (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 456–460.

[3] Christian Bizer, Peter Boncz, Michael L Brodie, and Orri Erling. 2011. The Meaningful Use of Big Data: Four Perspectives–Four Challenges. *SIGMOD Record* 40, 4 (2011), 57.

[4] Elena Cabrio, Julien Cojan, Alessio Palmero Aprosio, Bernardo Magnini, Alberto Lavelli, and Fabien Gandon. 2012. QAKiS: an open domain QA system based on relational patterns. In *International Semantic Web Conference, ISWC 2012*. CEUR-WS. org.

[5] CL Philip Chen and Chun-Yang Zhang. 2014. Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. *Information Sciences* 275 (2014), 314–347.

[6] Terry L Childers and Michael J Houston. 1984. Conditions for a picture-superiority effect on consumer memory. *Journal of consumer research* 11, 2 (1984), 643–654.

[7] Yuri Demchenko, Paola Grosso, Cees De Laat, and Peter Membrey. 2013. Addressing big data issues in scientific data infrastructure. In *Collaboration Technologies and Systems (CTS), 2013 International Conference on*. IEEE, 48–55.
[8] Diefenbach Dennis. 2017. WDAquaCore0Questions. https://github.com/WDAqua/WDAquaCore0Questions.
[9] Dennis Diefenbach, Shanzay Amjad, Andreas Both, Kamal Singh, and Pierre Maret. 2017. Trill: A reusable Front-End for QA systems. In *ESWC P&D*.
[10] Dennis Diefenbach, Andreas Both, Kamal Singh, and Pierre Maret. 2018. Towards a Question Answering System over the Semantic Web. *Semantic Web Journal (under review)* (2018).
[11] Dennis Diefenbach, Niousha Hormozi, Shanzay Amjad, and Andreas Both. 2017. Introducing Feedback in Qanary: How Users can interact with QA systems. In *ESWC P&D*.
[12] Dennis Diefenbach, Vanessa Lopez, Kamal Singh, and Pierre Maret. 2017. Core techniques of question answering systems over knowledge bases: a survey. *Knowledge and Information systems* (2017), 1–41.
[13] Dennis Diefenbach, Thomas Tanon, Kamal Singh, and Pierre Maret. 2017. Question Answering Benchmarks for Wikidata. In *ISWC 2017*.
[14] Dennis Diefenbach and Andreas Thalhammer. 2018. PageRank and Generic Entity Summarization for RDF Knowledge Bases. In *ESWC*. Springer.
[15] Marian Dörk, Sheelagh Carpendale, Christopher Collins, and Carey Williamson. 2008. Visgets: Coordinated visualizations for web-based information exploration and discovery. *IEEE Transactions on Visualization and Computer Graphics* 14, 6 (2008).
[16] David Ellsworth, Bryan Green, and Patrick Moran. 2004. Interactive Terascale Particle Visualization. In *Proceedings of the Conference on Visualization '04 (VIS '04)*. IEEE Computer Society, Washington, DC, USA, 353–360. https://doi.org/10.1109/VISUAL.2004.55
[17] Gartner. [n. d.]. IT Glossary – Big Data. Available from https://www.gartner.com/it-glossary/big-data/, Accessed 2018-05-20.
[18] Doug Laney. 2001. 3-D Data Management: Controlling Data Volume, Velocity, and Variety. 6 (01 2001).
[19] Teng-Yok Lee, Xin Tong, Han-Wei Shen, Pak Chung Wong, Samson Hagos, and L Ruby Leung. 2013. Feature tracking and visualization of the Madden-Julian Oscillation in climate simulation. *IEEE Computer Graphics and Applications* 33, 4 (2013), 29–37.
[20] James K Rayson. 1999. Aggregate towers: Scale sensitive visualization and decluttering of geospatial data. In *Information Visualization, 1999.(Info Vis' 99) Proceedings. 1999 IEEE Symposium on*. IEEE, 92–99.
[21] Tuukka Ruotsalo, Giulio Jacucci, Petri Myllymäki, and Samuel Kaski. 2015. Interactive intent modeling: Information discovery beyond search. *Commun. ACM* 58, 1 (2015), 86–92.
[22] Philip Russom et al. 2011. Big data analytics. *TDWI best practices report, fourth quarter* 19, 4 (2011), 1–34.
[23] Michael Schroeck, Rebecca Shockley, Janet Smart, Dolores Romero-Morales, and Peter Tufano. 2012. Analytics: The real-world use of big data. *IBM Global Business Services* 12 (2012), 1–20.
[24] Saeedeh Shekarpour, Edgard Marx, Axel-Cyrille Ngonga Ngomo, and Sören Auer. 2015. Sina: Semantic interpretation of user queries for question answering on interlinked data. *Web Semantics: Science, Services and Agents on the World Wide Web* (2015).
[25] Roger N. Shepard. 1967. Recognition memory for words, sentences, and pictures. *Journal of Verbal Learning and Verbal Behavior* 6, 1 (1967), 156 – 163. https://doi.org/10.1016/S0022-5371(67)80067-7
[26] Andrew Trotman. 2005. Learning to rank. *Information Retrieval* 8, 3 (2005), 359–381.
[27] Ryen W White and Resa A Roth. 2009. Exploratory search: Beyond the query-response paradigm. *Synthesis lectures on information concepts, retrieval, and services* 1, 1 (2009), 1–98.