Enabling Structured Data Generation by Nontechnical Experts

Andre Breitenfeld, Maximilian Mackeprang, Ming-Tung Hong,
Claudia Müller-Birn

Human-Centered-Computing, Freie Universität Berlin

Abstract

The Semantic Web provides meaning to information resources in the form of machine-accessible structured data. Research in the Semantic Web field focuses commonly on tools and interfaces for technical experts leading to various usability problems. The complexity of Semantic Web technology makes it difficult especially for nontechnical experts to use these technologies. Existing research on Semantic Web usability considers mostly consumers of structured data that leave out the creation perspective. In this work, we focus on the usability of creating structured data from text, especially on the creation of relations between entities. We reviewed existing research and state of the art annotation tools to establish shortcomings and used our knowledge to propose an interaction design for the creation of relations. We conducted a user-study which showed that the proposed interaction design performed well, making it a contribution to enhance the overall usability in the field.

1 Introduction

The Web has undergone two substantial evolvements Over the past decade, t that, initially, advanced almost in parallel. The first refers to web applications that create value based on user contributions, for example, people are editing articles in Wikipedia, sharing tagged photos on Flickr, or sharing source code on GitHub. The characteristics of these systems are summarized under the term Social Web. The second evolvement is based on the vision of a Semantic Web, in which structured data provides the meaning of existing information resources, such as websites, data or things. More recently, researchers have emphasized that both developments can complement each other: that the collective intelligence of the Social Web can augment the machine intelligence enabled by structured data. People can be both producers and conumeres of structured data in the so-called Social Semantic Web (Cahier and Zacklad, 2004). They are a source of knowledge for both, providing and using structured data for searching, querying or exploring (Lopez et al., 2012). Machines, on the other hand, are enablers, which aggregate and recombine structured data in a meaningful way. Even though researchers have taken up this vision with enthusiasm, a broad adoption is still missing. It has been argued that one reason for

Veröffentlicht durch die Gesellschaft für Informatik e. V. 2017 in
M. Burghardt, R. Wimmer, C. Wolff, C. Womser-Hacker (Hrsg.):
Mensch und Computer 2017 – Tagungsband, 10.–13. September 2017, Regensburg.
 https://doi.org/10.18420/muc2017-mci-0231

this low adoption is the lack of evaluation (Karger, 2014), which especially hinders nontechnical experts employing semantic tools. A finding from (Müller-Birn et al., 2015) on scholarly annotation practice shows that nontechnical experts, for example scholars who are experts in humanities, may hesitate to use more appropriate software due to an assumed steep learning curve. In the work presented here, we aim to close this gap by providing an interaction design that allows nontechnical experts to create structured data without prior knowledge of semantic technologies. This work's contributions are: (1) to highlight existing usability challenges in the adoption of semantic web technologies, especially in the field of semantic annotation; (2) to propose design rationales for an interaction design that abstracts the underlying data model and guides users in their annotation processes; and (3) to evaluate the proposed interaction design in a user study. In the first part, the paper describes the related work and states challenges with existing approaches. Section 3 details the interaction design that is derived from the challenges. Section 4 examines the usability of the proposed design based on a user study. Finally, we discuss implications in Section 5.

2 Theoretical Background and Related Work

A major goal of the Semantic Web is to achieve interoperability, enabling users to identify associated data across or integrate similar data from multiple distributed information resources. However, the "conventional" web has been designed for human consumption in the first place rather than for computer programs to manipulate meaningfully and process automatically. Semantic Web technologies tackles this problem by defining a technology stack that provides a standardized way of building meaning into data. In the following, we briefly introduce the selected technologies of this stack that are used to create the Semantic Web.

The Semantic Web Stack: The Semantic Web consists of structured data that states facts about any kind of thing or entity in the world that is stored in an information resource. These entities can be persons, places, etc. Each entity has a unique identifier. Properties are relations between resources. Statements assert the properties of resources, in other words, a resource has a property which consists of a value. This is expressed by a triple: "Subject — Predicate — Object" (SPO). A resource (subject) has a property (predicate) with a value (object). An object can be another resource or a literal (e.g. a number or a string). The fact, for example, that "John F. Kennedy was the 35th President of the United States" could be represented by the SPO: "John F. Kennedy" (S),"has position" (P),"35th President of the United States"(O)[1]. Triples can be used to describe semantic relationships which allows one to indicate whether relationships are symmetric (e.g. "If A isMarriedTo B" then this implies "B isMarriedTo A") or not.

Usability Concerns in the Semantic Web: Technical experts (e.g., ontology engineers) can use these technologies for standardizing data models and architectures, improving the performance of systems, or developing algorithms for automatic reasoning[2] (Carroll et al., 2004; Preist,

[1]This example simplifies the decomposition of the sentence in a SPO. The object "35th president of the United States" should be differentiated into additional statements separating role and number.

[2]Automatic reasoning refers to the ability to generate conclusions automatically using logical techniques such as deduction and induction from ontologies.

2004; Serafini and Tamilin, 2005). Many tools have emerged to support technical experts in these goals over the past fifteen years (e.g. Attwood et al., 2010; Gilson et al., 2008; Lohmann et al., 2016;). At the same time, users who are less technically minded are rarely considered in the development of semantic tools. Semantic Web technologies often require nontechnical experts to have prior knowledge of the technology (Dadzie et al., 2011). Research has identified reasons for the low usability of existing applications, for example, information overload due to the complex data model or technology involved (Jameson, 2006), differences between the mental model of the technical experts and nontechnical experts (Jameson, 2006), low interactivity (Dadzie et al., 2011), and a lack of usability studies (Di Maio, 2008).

Research that takes the user perspective into account considers users primarily as consumers of structured data. The needs of nontechnical experts using semantic technology applications address activities, such as searching, querying, exploring, or browsing (e.g. Dadzie et al., 2011; Jameson, 2006;Lopez et al., 2012). As mentioned before, users can also be producers, i.e. providers of structured data. This is especially important, since computer-understandable descriptions of resources are the backbone of the Semantic Web and generating high quality structured data cannot be done by machines alone (Cahier and Zacklad, 2004). Users need to define meaningful statements by annotating information resources, such as data or texts, to prepare structured data. The identification of resources (e.g., subjects) in existing annotation tools is often examined, but properties are considered more rarely. We describe semantic annotation and usability challenges of existing tools in the next section.

Usability of Semantic Annotation Tools: The term "annotation" refers to both the process of annotation and the result of that process (Oren et al., 2006). A semantic annotation as the result of such a process exhibits the following structure: (1) Entities, for example, the subject "John F. Kennedy" and the object "35th President of the United States" are parts of the text and (2) relations, for example, the predicate "has position" defines the type of relationship between the entities. An annotation can have different formality levels. Firstly, a semantic annotation can use URIs to identify resources. This is especially useful if, for example, a user refers to the entities "Kennedy", "JFK" or "John" in a text and we know it refers to the same person each time. Secondly, a semantic annotation can also be an ontological annotation, where the predicate is an ontological term, and the object conforms with an ontological definition (Oren et al., 2006). In our research, we refer to an ontological semantic annotation and focus on one part of the annotation process – the annotation of the relations. We conducted a state of the art review of existing semantic annotation tools. At first, we identified existing tools for semantic annotation based on existing studies and our own research. We identified 12 tools matching the criteria for the given work. Each of these tools was studied in an in-depth review. Due to space constraints, we omitted the in-depth reviews. Additional information of the tool selection criteria and all the results of the review are provided in a technical report[3].

The tool review included criteria such as the data abstraction chosen in the interface, the interaction pattern chosen for relation creation, the presence of user assistance, and the integration between the interface and the annotation target text. The results show that 7 out of the 12 tools used the SPO paradigm as the basis for the user interface and 4 introduced a kind of abstraction layer. When the SPO paradigm was chosen for the user interface, the interaction pattern most

[3] http://edocs.fu-berlin.de/docs/receive/FUDOCS_document_000000027360

used was a slot-filling approach (5 out of 7): The users had to select Subject, Predicate and Object from a drop-down menu to create the relation. This shows that the focus on technical users as described by literature is apparent in the absence of an abstraction layer over the data representation most of the tools (Dadzie et al., 2011; Jameson, 2006). Another finding was that most of the tools had no means of preventing annotating relations that conflict with the underlying semantic model. Only two of the tools leveraged constraints given by the relationship definition to prevent potential user errors. Furthermore, instead of providing users with quick cycles of action and evaluation (Jameson, 2006), some tools separated the text from the relation creation interface.

The last section showed a technical focus of Semantic Web technologies that leads to various deficiencies in the usability of tools from the Semantic Web area, as described by related work. Our own tool review confirmed these issues. Based on these insights, the next section proposes a new, visually focused interaction design that addresses these shortcomings.

3 Interaction Design for Relation Annotation

Based on the insights gained from the literature review and the results of the tool review, we derived four design rationales for our interaction design. These principles are introduced in the following.

Expose the User's Options: The proposed solution is a connection-centric approach rather than a select-source-and-target approach by point-and-click. Conceptually, the interface draws connections and suggests relations where appropriate. Only connections which are meaningful with respect to the underlying semantic model are visible for users. In the initial state of the interface, connections are hidden. Clicking on an entity enters the relation mode. The annotation selected will be visually connected to all meaningful entities. Entities which are not meaningful fade. (Pratt et al., 2010) state that animation attracts the user's visual attention. They found that objects involving animate motion were noticed more quickly than objects without. Using this principle, animation makes users aware of their options. Figure 1 shows the connection lines between two entities. Connecting lines are curved for aesthetic reasons. Furthermore, connections are selected by proximity. The connection with the smallest Euclidean distance to the cursor is selected (see highlighted connection in Figure 1). By selecting a connection, the subject and object of the relation are determined implicitly by the semantic model.

Focus the User's Intentions: In certain situations, the number of visible connections are too high, which could lead to an information overload (Jameson, 2006). We decided to minimize the cognitive load by limiting the number of visible connections to a visual range. This range is aligned to the user's area of interest. The area of interest corresponds to a user's gaze (Huang et al., 2012)[4]. Figure 2 illustrates how we used the cursor to construct the visual range. The center of the annotation selected and the cursor form a direction vector "v". The angle $alpha$ denotes the opening angle of the visual range. It is set to a fixed value of 10°. Only connections

[4](Huang et al., 2012) conducted a study on the gaze-cursor alignment. They concluded that there is a correlation between gaze and cursor position while the user examines and reads a web page. Furthermore, gaze-cursor alignment is particularly high if the user tries to target a visual element to perform an action.

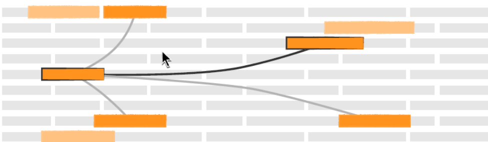


Figure 1: Connecting lines between entities.

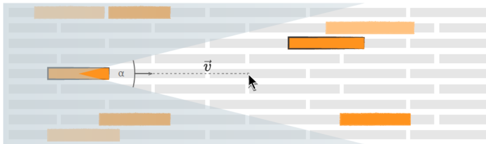


Figure 2: Reducing number of visible annotations by a limited field of view.

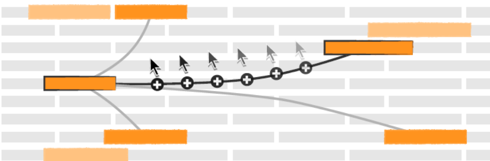


Figure 3: Movement of the Runner over time, when creating a new relation.

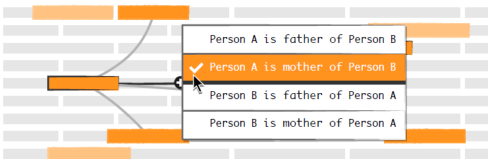
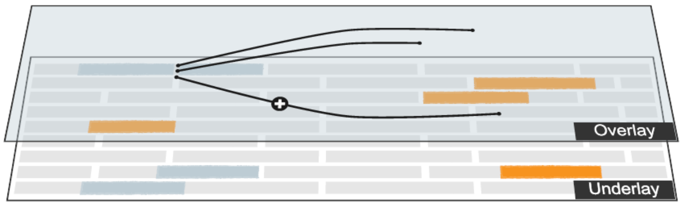


Figure 4: Statement menu on top of the Runner with annotation options with bidirectional support in natural language.

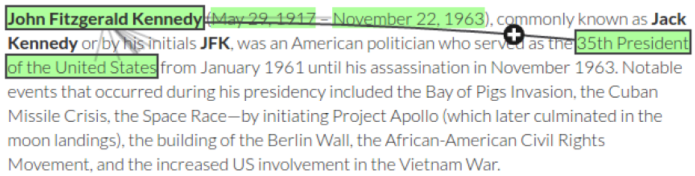
whose annotation anchors are inside of the field of view become fully visible. Connections whose annotations are outside are shortened (Figure 1). Existing relations are always visible to the user.

Accompany the User We introduce the concept of a Runner that is a visual element which follows the user's cursor along the selected connection path (cf. Figure 3). Fitts's law motivated the usage of a Runner, because selecting small objects takes a longer time (McGuffin and Balakrishnan, 2005). Connecting lines are thin objects and rather difficult to click on. In the light of Fitts's law, the Runner serves as a larger clicking target. The Runner has two states that decode whether a relationship exists on the connection selected or not. A plus icon indicates the absence of a relation and a pen icon indicates a present relation that can be modified. Clicking the icon on the Runner opens the statement menu, which is described as follows.

Speak the User's Language The statement menu is a context menu. It represents the last step to complete the definition of the relationship. We applied the principle of least astonishment to help nontechnical experts to understand their options, therefore, the statement is presented as a sentence in natural language. These statements are populated and composed grammatically based on the underlying semantic model, thus, only permitted relations between the entities selected are provided. All information about the entities involved and type and direction of the resulting relation are expressed as one sentence. Since it must be assumed that a relation exists already, the statement menu provides features to change or delete existing relations. A checkmark beside a statement denotes the existence of a relationship. In the case where a relation already exists, the statement menu shows the creator of the relation additionally. As described before, relationships can be symmetric, therefore, we introduced a bidirectional assistance (cp. Figure 4). Because of the bidirectional assistance the user does not need to consider the order of their interaction.



(a) Layers of the interface



(b) Screenshot from the overlapping result

Figure 5: Screenshot of the annotation tool used for the implementation of the proposed design.

4 Evaluation of the Proposed Design

We conducted a competitive within-subject usability study with 16 participants to evaluate the interaction design. The interaction design was implemented in the context of a web-based open source software for semantic annotation shown in Figure 5, to achieve a testable artifact. Consequently, the user interface component that is based on the javascript library AnnotatorJS was extended. The visual part was realized by an SVG overlay.[5] The participants in the study were researchers and students from local universities and research institutes. They were recruited by mailing list (13 male, 3 female). One participant was excluded from the study due to technical problems during the test session. The participants received no compensation for their participation. We provided a use case during the study in which participants were requested to generate structured data from textual resources. We told the test person that we must select a tool that is most appropriate for generating structured data from text for nontechnical experts. Even though most participants (13) indicated that annotating resources is an important part of their daily work, they had little knowledge of semantic annotation tools. We conducted a comparative usability study based on three tools to evaluate whether our proposed design outperforms existing approaches or not. One tool (T1) was representative of the slot-filling interaction paradigm. We chose one of the tools providing an abstraction layer for the second tool (T2) . The third tool (T3) was our own implementation.[6] A within-subjects design was employed following the recommendation of literature on competitive studies (Goodman et al., 2013). The order of the tools presentation was randomized to mitigate familiarization effects. The study was originally conducted in German. The questions and qualitative results presented subsequently were carefully translated for publication.

[5] The complete source code is available here: https://github.com/FUB-HCC/neonion

[6] These tools were selected because of the results of the in-depth tool review described in Section 2. Further selection criteria were binary/source distribution availability, text layout similarity and the availability of an English localization. T1 is the tool Pundit (Grassi et al., 2013) and T2 is the tool Glozz (Widlöcher and Mathet, 2012).

We focused our evaluation on the following criteria in the usability study: learnability ($C_{\text{learnability}}$), satisfaction ($C_{\text{satisfaction}}$) and efficiency ($C_{\text{efficiency}}$) to consider especially the requirements revealed in previous research (Dadzie et al., 2011; Di Maio, 2008; Jameson, 2006). We divided the study into two phases. In the first phase, i.e. the learning phase (P_{learning}), we assessed the learnability of the interfaces provided. After the first phase, we explained all interfaces to the participants and, thus, prepared the second phase, i.e. the working phase (P_{working}). The satisfaction and efficiency have been evaluated in both phases. Learnability and satisfaction were determined by a questionnaire (Q1: "How easy was it to learn the tool after the first-time use?"; Q2: "How much do you like the tools after first use?"; Q3: "How much do you like the tools after you've worked with them?"). Because we carried out a competitive test design, test participants were asked to order the tools from easy to use thgough neutral to difficult to use. Efficiency was measured by task completion. We encouraged participants to verbalize their thoughts while using the tools (Thinking-Aloud) to receive qualitative feedback. Our task design was motivated by previous research indicating that annotating texts is a foundational activity in humanities (Unsworth, 2000). Since the focus of the solution designed is on the relation annotation, we provided participants with a text entities already annotated, asking them to create relationships between them. One relationship had to be created in the learning phase. The task had to be completed without prior explanation of the tools. We set a time limit of three minutes for this task and tracked completion of the task to evaluate the efficiency of the design. After completing the task, participants were asked to rank the tools in terms of their learnability and satisfaction. Afterwards, they received a detailed explanation of the tool's functionality. The working phase consisted of six different annotation creation tasks. The difficulty of the tasks was similar except for the last task where users had to create a symmetric relationship. The time limit for this phase was set at five minutes. After completing the working phase, the participants were asked to fill out another questionnaire.

4.1 Results

Table 1 shows the results of our study for each phase and tool. Each row shows the rankings picked by the relative amount of people for each tool (e.g. the first cell shows that 26 % of the participants chose T1 as the best ranking tool in terms of ease of use). Furthermore, we added the task completion rates for a concise reference to all results. We explain these results in more detail in the following. After the learning phase, 73 % of the participants found the proposed interaction design was the easiest in terms of learnability (Q1) and again 73% of the participants found the proposed interaction design most satisfactory (Q2) ($p = 0.0017$). The task completion rate for both, the proposed design and the slot filling design (T1) was very high, as opposed to the point-and-click design provided by T3, where only 53% of the testers could to complete the task. All participants completed all tasks within the given time limit (100% task completion rate) in the working phase. Even though user satisfaction is still the highest for the proposed design, the difference in the results is less high. The proposed interaction design was most satisfactory for 60 % of the test participants ($p = 0.03$), but 20% of the participants were highly satisfied for each of the other two tools [7].

[7] P-values were calculated using the null-hypothesis that participants filled out questionnaires randomly.

Phase	Criteria	Tool	T1 (slot-filling)			T2 (point-and-click)			T3 (proposed design)		
		rank	1^{st}	2^{nd}	3^{rd}	1^{st}	2^{nd}	3^{rd}	1^{st}	2^{nd}	3^{rd}
P_{learning}	$C_{\text{learnability}}$	Q1	26%	66%	6%	0%	6%	93%	73%	27%	0%
P_{learning}	$C_{\text{satisfaction}}$	Q2	26%	53%	20%	0%	20%	80%	73%	27%	0%
P_{learning}	$C_{\text{efficiency}}$	TC*	92%			53%			100%		
P_{working}	$C_{\text{satisfaction}}$	Q3	20%	60%	20%	20%	0%	80%	60%	40%	0%
P_{working}	$C_{\text{efficiency}}$	TC*	100%			100%			100%		

*Table 1: The percentage of ratings for each tool (T1, T2, T3) is provided (n = 15). Each rank position sums up to 100 %, for example, 20 % of the participants ranked T1 as best in terms of satisfaction (Q3), while 20 % ranked T2 best and 60% ranked T3 as best. * TC refers to the task completion rate for each phase and tool.*

4.2 Discussion of Results

The completion rate and the high ratings for both satisfaction and learnability in the questionnaire of the learning phase suggest that the proposed design is more intuitive and easier to learn than the other two designs. The proposed design provides an interface for nontechnical experts to build up their mental models. Ot bridges the gap of mental models between technical and nontechnical experts as proposed by (Jameson, 2006). We conjecture that the proposed design provides more assistance to the user to create structured data by annotating text semantically. The assistance hides the complexity of semantic technology involved successfully, as suggested by (Dadzie et al., 2011). However, the results are less conclusive, after the working phase. The trade-off between learnability and efficiency of a software system especially could not be evaluated due to the 100 % completion rate for all three tools. One reason for the high completion rate might be the low difficulty of the task provided. The text used in the user study was rather simple and the wording in the task was quite similar to the concepts of the ontology. The tool with the proposed design needs to be evaluated in a field study to evaluate efficiency. We suppose that the lack of an immediate feedback in the proposed design after creating a relation is a potential source for confusion. The change of the Runner icon from a plus-icon to a pen-icon does not provide enough information whether a relationship was saved or not. The thinking aloud protocols revealed that multiple participants were confused by this fact (P3, P4, P6). Moreover, a further analysis of the thinking aloud protocols showed that the bidirectional assistance was also not easy for participants to use. One participant (P2), for example, accidentally added a relation by mixing up the direction of the statement. Instead of annotating "A was killed by B", the person created the annotation "B was killed by A". Even though participants were encouraged to read all statements provided by the Runner carefully, it did not prevent user errors. Further improvement of the presentation of the possible statements regarding the Runner are needed. Another unexpected result was the low task completion rate during the learning-phase for T2 as a representative of the group of tools providing an abstraction. One possible explanation for this outcome is that the creation of relationships is not the default mode used in the tool. Qualitative feedback by multiple participants (P1, P3) indicates that users did not know how to switch into the relation-creation mode without an explanation.

Limitations The usability evaluation was subject some limitations. One is the selection of the tasks: The evaluation was focused on annotating relations, therefore entities (subjects, objects) were already annotated in each tool. Thus, even though the usability of annotating relationships was considered as satisfactory and easy-to-learn, the results could be affected by annotating entities as well as relationships. The terminologies defined in the tools were prepared for the user tasks and contained only the useful semantic relationships of classes and properties.A larger terminology would be used in a more realistic scenario. Consequently, the importance of the proposed user assistance grows. We could only install a subset of tools in our test setup; therefore, we had to limit our study to one representative tool for each interaction paradigm. This limits the generalization of our study results.

5 Conclusion

In this paper, we propose a new interaction design that allows nontechnical experts to create structured data from text by means of semantic annotation. Based on existing research and a tool review, we derive design rationales for a new interaction design that was realized and evaluated in a software tool. Our results indicate that the task of creating relationships between semantic annotations can be improved in terms of learnability and satisfaction in comparison to state of the art semantic annotation tools by using the proposed interaction design. This suggests that a more user-centric approach to content creation in the context of Semantic Web applications could increase adoption by nontechnical experts and, therefore, might expand the overall use in both research and commercial applications. Due to the highly specialized area of semantic relations, our design and evaluation only supports a small set of content generation tasks in the area of the Semantic Web. Further research could expand the use cases supported, for example, by providing an additional layer to web content to facilitate the vision of the Social Semantic Web.

Acknowledgements

We gratefully acknowledge the support of the Elsa-Neumann-Stipendium of the state Berlin, and the team of Hybrid Knowledge Interactions at the Cluster of Excellence Image Knowledge Gestaltung: An Interdisciplinary Laboratory (EXC 1027).

The authors acknowledge the financial support by the Federal Ministry of Education and Research of Germany in the framework of Ideas to Market (project number 03IO1617).

References

Attwood, T. K. et al. (2010). Utopia documents: Linking scholarly literature with research data. *Bioinformatics*, *26*(18), i568–i574.

Cahier, J.-P. & Zacklad, M. (2004). "Socio-Semantic Web" applications: Towards a methodology based on the theory of the communities of action. *resource*, *8*(400).

Carroll, J. J. et al. (2004). Jena: Implementing the semantic web recommendations. In *Proc. of the WWW* (pp. 74–83). ACM.

Dadzie, A. S., Rowe, M., & Petrelli, D. (2011). Hide the stack: Toward usable linked data. In *Lecture notes in computer science* (pp. 93–107).

Di Maio, P. (2008). Toward global user models for semantic technologies: Emergent perspectives. In *Proc. of ASWC* (Vol. 8, pp. 141–152).

Gilson, O., Silva, N., Grant, P. W., & Chen, M. (2008). From web data to visualization via ontology mapping. In *Computer graphics forum* (Vol. 27, *3*, pp. 959–966).

Goodman, E., Kuniavsky, M., & Moed, A. (2013). Observing the user experience: A practitioner's guide to user research. *IEEE TPC*, *56*(3), 260–261.

Grassi, M. et al. (2013). Pundit: Augmenting web contents with semantics. *Literary and Linguistic Computing*, *28*(4), 640–659.

Huang, J., White, R., & Buscher, G. (2012). User see, user point: Gaze and cursor alignment in web search. In *Proc. of the sigchi conference* (pp. 1341–1350). ACM.

Jameson, A. (2006). Usability and the Semantic Web. In *ESWC* (pp. 3–3). Springer.

Karger, D. R. (2014). The semantic web and end users: What's wrong and how to fix it. *IEEE Internet Computing*, *18*(6), 64–70.

Lohmann, S., Negru, S., Haag, F., & Ertl, T. (2016). Visualizing ontologies with vowl. *Semantic Web*, *7*(4), 399–419.

Lopez, V. et al. (2012). PowerAqua: Supporting users in querying and exploring the Semantic Web. *Semantic Web*, *3*(3), 249–265.

McGuffin, M. J. & Balakrishnan, R. (2005). Fitts' law and expanding targets: Experimental studies and designs for user interfaces. *TOCHI*, *12*(4), 388–422.

Müller-Birn, C. et al. (2015). Neonion: Combining human and machine intelligence. In *Proc. of Companion on CSCW*. ACM.

Oren, E., Möller, K., Scerri, S., Handschuh, S., & Sintek, M. (2006). What are semantic annotations. *Relatório técnico. DERI Galway*, *9*, 62.

Pratt, J., Radulescu, P. V., Guo, R. M., & Abrams, R. A. (2010). It's alive! animate motion captures visual attention. *Psychological Science*, *21*(11), 1724–1730.

Preist, C. (2004). A conceptual architecture for semantic web services. In *ISWC* (pp. 395–409). Springer.

Serafini, L. & Tamilin, A. (2005). Drago: Distributed reasoning architecture for the semantic web. In *ESWC* (pp. 361–376). Springer.

Unsworth, J. (2000). Scholarly primitives: What methods do humanities researchers have in common, and how might our tools reflect this? In *Symposium on humanities computing: Formal methods, experimental practice. king's college, london* (Vol. 13, pp. 5–00).

Widlöcher, A. & Mathet, Y. (2012). The glozz platform: A corpus annotation and mining tool. In *Proc. of the doceng* (pp. 171–180). ACM.

Author Information

Breitenfeld, Andre

Andre Breitenfeld has recently completed his M.Sc. in Computer Science at the Freie Universität Berlin. He is interested in interaction and service design. His current work focuses on digitalization and mobility services in the automotive industry.

Mackeprang, Maximilian

Maximilian Mackeprang has recently completed his M.Sc. in Computer Science at the Humboldt Universität zu Berlin. His current research interests include Human-Centered Design, Usability and Collaborative Ideation Software.

Hong, Ming-Tung

Ming-Tung Hong is a PhD student in Computer Science at the Freie Universität Berlin. Her research focuses on augmenting humans' sensemaking processes based on semantic annotations by allowing humans to interact with machine-recommended information. She is interested in research areas such as Computer-Supported Cooperative Work and Intelligent User Interfaces.

Müller-Birn, Claudia

Prof. Dr. rer. nat. Claudia Müller-Birn is the head of the research group Human-Centered Computing (HCC.lab) at the Institute of Computer Science at the Freie Universität Berlin. Her research advances the fields of Computer-Supported Cooperative Work and Social Semantic Computing.