# Semantic Annotation and Information Visualization for Blogposts with *refer*

Tabea Tietz[1], Joscha Jäger[2], Jörg Waitelonis[1], and Harald Sack[1]

[1] Hasso-Plattner-Institute, Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, Germany
`firstname.lastname@hpi.de`
[2] yovisto GmbH, August-Bebel-Str. 26-53, 14482 Potsdam, Germany
`firstname@yovisto.com`

**Abstract.** The growing amount of documents in archives and blogs results in an increasing challenge for curators and authors to tag, present, and recommend their content to the user. *refer* comprises a set of powerful tools focusing on Named Entity Linking (NEL) which help authors and curators to semi-automatically analyze a platform's textual content and semantically annotate it based on Linked Open Data. In *refer* automated NEL is complemented by manual semantic annotation supported by sophisticated autosuggestion of candidate entities, implemented as publicly available Wordpress plugin. In addition, *refer* visualizes the semantically enriched documents in a novel navigation interface for improved exploration of the entire content across the platform. The efficiency of the presented approach is supported by a qualitative evaluation of the user interfaces.

**Keywords:** visualization, annotation, named entity linking, DBpedia

## 1 Introduction

When searching for an arbitrary subject in weblogs or archives, users often don't get the information they are really looking for. Often they are overwhelmed with an overflow of information while sometimes the presented information is too scarce to make any use of it. Without further knowledge about the context or background of the intended subject users are easily frustrated because they either cannot handle the amount of information or they might give up because they cannot make sense of the topic at all. Furthermore, authors of online-platforms often deal with the issue to provide useful recommendations of other articles and to motivate the readers to stay on the platform to explore more of the available but most times hidden content of their blog or archive. With *refer* users are encouraged to take an active part in discovering a platform's information content interactively and intuitively, rather than just to have to read the entire textual information provided by the author. They can discover background information as well as relationships among persons, places, events, and anything related to the subject in current focus and are inspired to navigate the previously hidden information on a platform.

To enable content discovery in blogs and archives, *semantic annotations* are used to enrich texts with additional information to explain relations among entities as well as to provide meaningful content-based recommendations. However, common issues with (semantic) annotations are that their provision and maintenance is an extra effort to handle and lay-users find it rather difficult to deal with Linked Data [1]. With *refer*, content creators are enabled to (semi-)automatically annotate their text-based content with DBpedia resources as part of the original writing process and visualize them automatically. Thereby, authors can engage the readers to further explore the available content and to provide background information from DBpedia and Wikipedia without having to leave the platform.

In this paper the newly developed user interfaces of *refer* for semantic annotation and visualization together with a qualitative evaluation are presented. The goal of the evaluation is to better understand how to display entities in semantic annotation interfaces in order to support lay-users to annotate text as completely, accurately, and conveniently as possible. A preliminary user study on the proposed visualization interfaces to explore the annotated content was performed with the intention to receive insights on how to display the information to actually provide valuable additional content without overwhelming the user.

The original concept of the presented user interface and a first prototype have already been presented in [8]. This paper focuses on the achieved improvements based on two years of user experience with the implementation of the working system. The contributions of this paper include the implementation of new annotation and visualization interfaces as Wordpress-plugin, a proof-of-concept by integrating the system into a daily weblog[3], the Wordpress-plugin of *refer* publicly available for download[4], supported by a detailed user study on the proposed Linked Data annotation techniques as well as a preliminary user study on the three proposed Linked Data visualizations. All data gathered during the evaluation process is publicly available for further use[5].

The paper is structured as follows. In Section 2, related annotation and visualization systems, as well as alternative scientific approaches are discussed, followed by a detailed description of all *refer* components in Sect. 3. Sect. 4 presents the evaluation of the proposed user interfaces and the achieved results are discussed in Sect. 5. A conclusion and outlook on future work is provided in Sect. 6.

## 2 Related Work

Dadzie and Rowe [1] provided an in depth survey on various Linked Data visualization and exploration techniques. In this section, more recent approaches and techniques similar to *refer* are discussed and compared against our approach.

---

[3] http://blog.yovisto.com
[4] http://refer.cx/
[5] http://s16a.org/refer

Trinh et al. have proposed an autocomplete input box for manual semantic annotations, developed for content creators to provide readers with additional meta-information about the content [9]. *refer* combines automated and manual annotation to improve the annotation quality. Furthermore, in this paper also the visualization of enriched information is evaluated.

One major goal of *refer* is to enable users to explore the content of a platform actively. However, content consumers are not required to provide annotations as e.g. with Pundit [6]. *refer* leaves this task to the content authors for the following reasons: (1) Annotation quality: being an authoritative source the author knows best about information context and pragmatics. Thus, author annotations are considered as being more accurate and being provided faster and with less effort. (2) Abuse: website hosts do not have to take care of the potential abuse of the annotation interface by malicious users. (3) Linked Data complexity: lay-users often find it difficult to work with Linked Data and might easily give up when semantically annotating content written by another author. Even though most authors will also be lay-users (regarding Linked Data), they have profound knowledge about their provided content and will be able to learn how to correctly annotate their texts much faster. Pundit allows users to choose and define their own properties and knowledge bases for the annotation. However, *refer's* target users are not only journalists, professionals, and researchers, but everyone who creates content on the web, which may include travel blogs as well as websites about cooking or fashion platforms. *refer* offers less complexity and only requires the authors to annotate content with DBpedia entities.

The Poolparty thesaurus[6] plugin for Wordpress imports a SKOS thesaurus or a thesaurus from a public SPARQL endpoint and automatically links terms in blog posts to the thesaurus. On mouse-over, definitions of the linked entities are provided. The KEA named entity linking (NEL) tool [11] as part of *refer* annotates texts with DBpedia entities automatically and is supported by manual override to keep control of the content completely with the authors. In addition to the showing definitions on mouse-over as in the Poolparty thesaurus, *refer* also visualizes links to related entities in DBpedia and recommendations.

WYSIWYM by [3] is an approach for integrated visualization, exploration and authoring of unstructured and semantic content. As part of the approach, the authoring tool RDFaCE [4] and the conTEXT [5] interface were implemented, which are similar to *refer*. However, *refer* was optimized especially for lay-users and thus has to cope with different challenges compared to [3], such as, e.g. how to display semantic information to be useful for non-experts in the authoring and exploration environment. Furthermore, our evaluation differs from [3] because it includes more participants outside the academic and computer science domain.

## 3   *refer* Components and Infrastructure

The *refer* system consists of the following tools and visualizations, which are integrated into Wordpress. The *Annotator* is an extension of the text editing

---

[6] `https://wordpress.org/plugins/poolparty-thesaurus/`(accessed: June 29, 2016)

interface to create semantic text annotations based on DBpedia. *Infoboxes* are used to visualize annotations in the article view. The *Relation Browser and Recommender* visualize relationships between annotations as well as suggestions for further reading.

### 3.1    Annotation Tools

To create annotations, the author selects a text fraction ranging from a single word to the entire article text in the Wordpress editing environment. The author can further choose between manual and automated annotation by means of automated NEL, which is the task of computationally determining the identity of entities mentioned in the text by linking the textual mention to a knowledge-base entity. Thereby ambiguous textual mentions are algorithmically disambiguated by analysing the context of the mentions. For automated annotation, *refer* deploys the KEA-NEL [12], which implements entity linking with DBpedia entities [10]. Nevertheless, there are still errors that have to be revised via manual annotation. The user has to be supported in selecting the correct entity from the large knowledge-base, which is especially difficult for highly ambiguous textual mentions. For example, for the term 'Michigan', e.g.`dbp:Lake_Michigan`, `dbp:Michigan_wine`, `dbp:Michigan`, and many more can be considered as potential candidates. Some entity mentions yield to lists of thousands of candidates
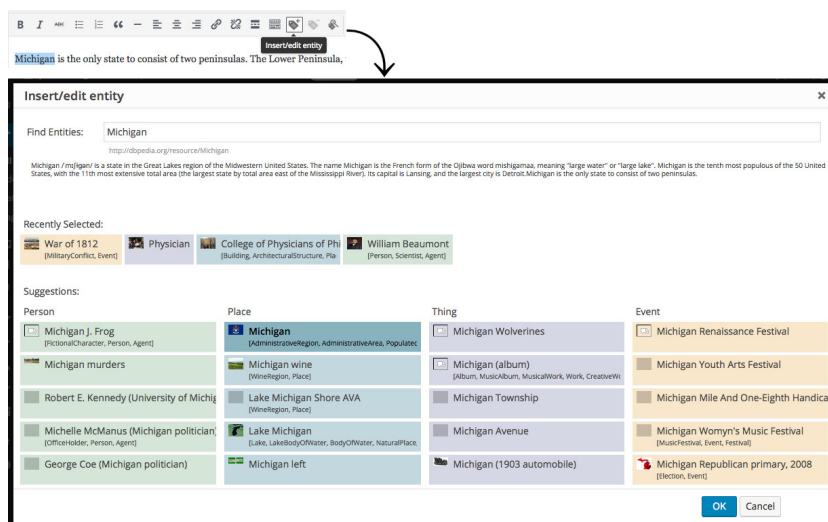


**Fig. 1.** Modal Annotator

which a human cannot survey quickly to find the correct one. Therefore, *auto-suggestion* utilities are applied to rank and organize the candidate lists according

to e. g. string similarity with the entity mention, or popularity of the entity [7]. But also the visual presentation of suggestions determines the users' annotation performance. To support this claim, we introduce and evaluate two different visual presentations of autosuggestion for text annotation.

The *refer* Annotator provides two configurable user interface modes: *modal* and *inline*. The **Modal Annotator** (see Fig. 1) builds upon the native TinyMCE editor controls (part of the Wordpress installation) to trigger the display of suggested entities in a modal dialog window. Upon text selection, the user can choose to open the suggestion dialog or automatically scan the selected text for entities via new buttons in the TinyMCE control panel. Entities added to the text either via manual or automated annotation can always be edited or removed by the user via a context-menu located right beside each entity in the text. The suggestion dialog starts with a text input field, which initially contains the selected text fragment and can be used to refine the search term. Suggested entities are shown below in a table-based layout, divided into the four categories Person (green), Place (blue), Event (yellow) and Thing (purple), including a list of recently selected entities for faster selection of already annotated entities in the same text. A suggested entity is displayed by its label, thumbnail, and main categories. The text abstract and entity IRI are displayed on mouseover. The selected annotation is encoded in RDFa markup, which is added to the according text fragment.
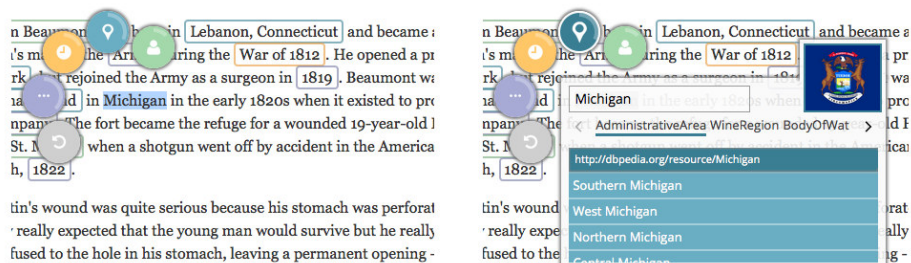


**Fig. 2.** Inline Annotator

The **Inline Annotator** (see Fig. 2) enables to choose entities directly in the context of a selected text. The basis of the inline annotation solution is a circular category menu attached to a text fragment upon selection and allows the user to instantly show suggestions from the respective category (Person, Place, Event, Thing). Additionally, a list of recently selected entities from all categories can be displayed. By selecting a category, the suggested entities are displayed. In order to provide more context within the relatively small space, these entities are divided into dynamically retrieved sub-categories, which are rendered horizontally as navigable tabs and are based on the list of categories

per entity provided by the DBpedia type system. The rationale of the Inline Annotator is to provide fast and simple means of semantic text annotation by minimizing the steps required to open the interface, visually scan the suggestions in several categories and to choose the most appropriate entity. Compared to the modal annotation interface, the Inline Annotator integrates directly into the text area, requires less space and preserves the context of the annotated text fragment. By combining the interactions required to open the suggestion menu and choose a category, the user is able to choose an entity more quickly. On the other hand, the modal interface leaves more space for annotations and additional information, and provides a parallel view of all categories.

### 3.2 Infobox Visualization

Annotated entities are indicated directly in the article text. To avoid disrupting the reading flow and visual design of the surrounding webpage, entities in the text are visualized by thin, semi-transparent, colored lines below the respective fragments. The color code indicates the same four categories (Person, Place, Event, Thing) as in the annotation interfaces. On mouseover, an infobox as in Fig. 3 is shown right below the annotated text fragment, which contains basic information about the entity, e.g. a label and thumbnail as well as additional data in a table-layout. The visual design and content of infoboxes varies per category and allows the user to gather basic facts about an entity as well as relations to other entities. While some basic information



**Fig. 3.** Infobox

can be derived just from the webpage's RDFa microdata and is displayed instantly, additional content is asynchronously loaded from the web service once the infobox is shown for the first time. When the text fragment or any of the infobox entities are clicked, the *Relation Browser* slides down from the top of the page with the selected entity in focus.
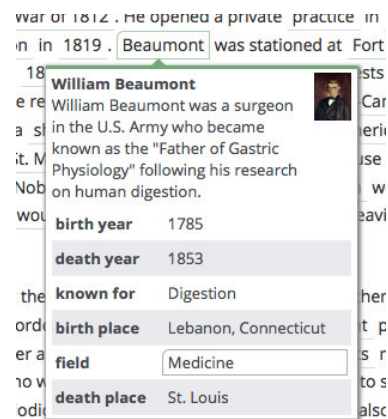
### 3.3 Relation Browser and Recommender

The *Relation Browser* (cf. Fig. 4) allows users to navigate and explore relations among entities. It can be opened at any time by the user either via click on the *refer* icon bar on top of the page or by selecting an entity in the article text. The rationale here is that if a user is interested in an entity annotated in the text, (e.g.
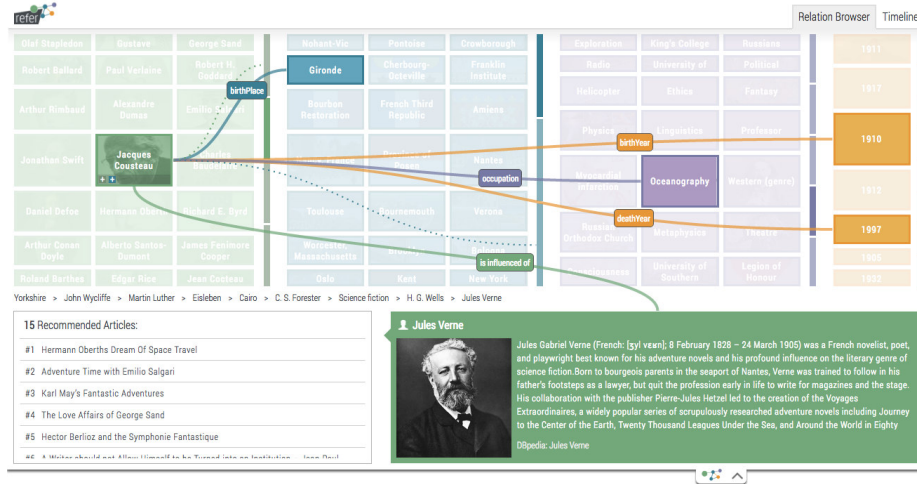
**Fig. 4.** Relation Browser with entity Jules Verne in focus and the Recommender on the bottom left

*Jules Verne*), she clicks on the entity in the text and thereby opens the Relation Browser. The entity *Jules Verne* then becomes the focus-entity. Its DBpedia abstract and image are displayed on the bottom of the Relation Browser and its background color depends on its category (Person, Place, Event, Thing). Based on the focus-entity, related entities (derived from DBpedia and all annotations available on the platform) are displayed in a four column grid.

On the right hand side of each column, pagination bars indicate the amount of further entities available within each category. Clicking the bars allows the users to browse through all entities in a category. When hovering one of the displayed entities (e.g. *Jacques Cousteau* in Fig. 4), relations to the focus-entity (e.g. *Jules Verne*) and to further entities in the grid-view (e.g. *Oceanography*) are visualized by line connectors. A label (property) indicates the direction and type of connection. If there are more entity-relations than displayed in the first overview, connections to hidden entities are indicated by dotted lines, which can be activated via hovering a small 'plus' icon inside the entity box.

A click on an entity in the grid-view replaces the focus-entity (*Jules Verne*) with the selected item and refreshes the related entities in all categories. A ranked list of recommended blogposts for the entity in focus is displayed on the bottom-left. The recommendations comprise blogposts that cover the focus-entity as well as entities related to the focus entity. The more entities are related with the entity in focus, the higher is the rank of the recommended article in the list. A recent survey on sophisticated Linked Data based recommender systems is given in [2].

## 4 Evaluation

A qualitative user study was performed to evaluate both annotation interfaces (Modal and Inline) as well as the visualization interfaces (Infobox, Relation Browser, Recommendation). In total, 20 participants took part in the study, aged between 21 and 45. Half of the users have a background in computer science, the others in various domains, such as teaching, biology, engineering, sports, marketing, beauty, and design including participants from the non-academic field as well.

Only 5 participants considered themselves experts with Linked Data technologies while 11 test-users had either no prior knowledge about Linked Data or had only heard about it before. All participants use the Web several times a day. Since all test-users are German native-speakers, the experiment has been performed in German language, while the user interface and annotated texts have been presented in English. Therefore, the test users had to be fluent in the English language. For each participant the experiment lasted 40 to 50 minutes and took place in a controlled environment with one interviewer present, who took notes on the participants' comments as well as their annotation and navigation behavior. The evaluation covered two parts. First, the participants were asked to annotate two consecutive text snippets with one annotation interface each. Second, the users had to solve specific tasks given in the navigation and exploration environment.

All survey sheets and evaluation results are available for download.[7]

### 4.1 Semantic Annotation

To find out which features are most helpful to annotate text with DBpedia entities, both annotation interfaces were tested for usability and accuracy. After a short introduction, each participant received a text paragraph containing a variety of entity-types, including persons, dates, events, places, and common nouns. Moreover, the text includes terms for which the users had to highly focus on the context of the sentence in order to disambiguate correctly. The paragraphs and interfaces alternated for each user, who annotated one text with each interface. After reading the presented paragraph, the participants were told to annotate the text as accurate, as complete, and as specific as possible. Specific in this context means that e.g. in the case of the compound *John F. Kennedy Airport*, the entire term should be annotated with one DBpedia entity `dbp: John_F._Kennedy_International_Airport` [8] instead of `dbp:John_F._Kennedy` and `dbp:Airport` separately. For each annotation task, the interviewer measured the required time. Next, the participants completed a short survey and an open interview was performed after both annotation tasks were finished. All questions concerned the understandability, readability, ease and fit of use, the ease of learning, and subjective speed and accuracy of both interfaces. A ground truth

---

[7] `http://s16a.org/refer`
[8] The prefix `dbp:` stands for `http://dbpedia.org/resource/`

containing correct annotations for both texts has been published previously [11] and was used to measure the annotation accuracy of all participants. The evaluation further helped to categorize common mistakes made by the users to optimize the interface in future work.

### 4.2 Navigation and Exploration

The second part of the user study covered all visualization and exploration interfaces. The goal was to find out how semantic information should be displayed in the context of a blogpost to make sure the enriched information is actually useful and does not overwhelm or distract the participants. As starting point of this study served an already annotated article[9]. Each user was asked 11 questions to be answered orally, including:

- What is Michael Polanyi best known for?
- How is Eugene Wigner connected with Technical University Berlin?
- Which blogpost can be recommended for the year 1902?[10]

Most answers cannot be found in the available article text, but via navigating the interface. While the participants were searching for the correct answers, the interviewer took notes on how the participants attempted to achieve the information of interest. After the task was finished the participants again completed a survey.

## 5 Results and Discussion

In this section, all achieved results are presented in detail followed by an in depth discussion.

### 5.1 Annotation Interfaces

Table 1 depicts the relative scores calculated from the Likert-type survey each user completed after using each annotation interface along with the average annotation duration per paragraph. While the participants found that the modal annotation interface was slightly easier to learn and both interfaces received the same score in terms of understandability, the inline annotator is valued slightly better in all the remaining categories.

As Table 1 shows the inline annotator just slightly achieved better results, therefore we also took into account the comments on both interfaces provided by the participants on the survey sheets and orally. Thereby, it became clearer that the inline annotator was favored by most participants in terms of usability. The participants felt that annotations can be made faster, due to its size the context of the paragraph was still available, and the interface was triggered automatically

---

[9] http://blog.yovisto.com/?p=9
[10] The complete questionnaire is available at http://s16a.org/refer

|  | Inline Annotator | Modal Annotator |
|---|---|---|
| Understandability | 0.86 | 0.86 |
| Readability | 0.91 | 0.86 |
| Learnability | 0.97 | 0.98 |
| Usability | 0.86 | 0.87 |
| Utility | 0.79 | 0.77 |
| Subjective Accuracy | 0.86 | 0.84 |
| Subjective Speed | 0.94 | 0.9 |
| Average Duration (mm:ss) | 06:04 | 07:12 |

**Table 1.** Relative usability scores retrieved from the Likert-type questions

instead of having to click on a button to initiate entity suggestion. On the other hand, some still favored the modal interface because it provided a more complete overview of all entity categories as well as short entity descriptions. In order to measure whether one of the interfaces enabled more accurate annotations, the results from all participants are compared to the ground truth and to the automated annotation via KEA-NEL. Table 2 shows that the modal annotation interface enabled the users to annotate more accurately by 3% F-measure. Both interfaces have almost the same recall at ca. 68-69%, meaning that about 31% of annotations are missing. Consequently, the modal annotation interface exhibits a better precision (+5%). The KEA-NEL found every annotation with a recall of 100%, but the precision drops below 60%.

|  | Precision | Recall | F-measure |
|---|---|---|---|
| Inline | 0.826 | 0.676 | 0.752 |
| Modal | 0.882 | 0.693 | 0.788 |
| KEA-NEL | 0.582 | 1 | 0.791 |

**Table 2.** Comparison of annotation accuracy between both interfaces and KEA-NEL

|  | Inline | Modal | Total | KEA-NEL |
|---|---|---|---|---|
| Missing | 0.64 | 0.66 | 0.65 | 0 |
| Compound Split | 0.13 | 0.13 | 0.13 | 0.10 |
| General/Specific | 0.13 | 0.12 | 0.12 | 0.10 |
| Wrong Entity | 0.11 | 0.10 | 0.10 | 0.81 |

**Table 3.** Relative occurrence of all error-categories regarding both annotation-interfaces, overall manual annotations, and automated annotations by KEA-NEL.

All errors resulting from the manual and automated annotation have been manually classified into predefined error categories (cf. Table 3). The goal was to identify the most and least common mistakes in both interfaces which might be resolved by improving information arrangement in future versions of the interfaces. Four different error-categories have been identified: (1) Missing: terms which have not been annotated, but should have according to the ground truth. (2) Compound Split: entities such as e.g. `dbp:Nobel_Prize_in_Physics` which have been split into two separate entities, as e.g. `dbp:Nobel_Prize` and `dbp:Physics`. (3) General vs. Specific: terms for which a more general entity has

been chosen instead of a more specific one as required by the ground truth, e.g. `dbp:Army` instead of `dbp:United_States_Army`. (4) Wrong Entity: wrongly annotated entities not classified in category 1-3, such as e.g., `dbp:Michael_Polanyi_Center` as a location instead of `dbp:Michael_Polanyi` as a person. Table 3 shows that the most common mistakes for manual annotations belong to category (1), which was the least common mistake for the automated KEA-NEL and also reflects the recall-result in Table 2. Vice versa, category (4) was calculated as the least common mistake for the human annotators while it was the most frequent KEA-NEL error. In both interfaces, about 13% of all errors have been classified as a compound split error and 12% of all errors have been made because the users have chosen too general entities. Table 3 shows that humans tend to make category (2) and (3) errors, which shows that users might have difficulties to recognize compounds and specific words as annotation subject. Interestingly, the KEA-NEL detects this language phenomenon more accurately, which might be caused by the optimization of text analysis in a top-down fashion, preferring larger text fragments over single words [12]. Contrarywise, in the actual disambiguation process KEA-NEL produces much more erroneous annotations than the users. In conclusion, it seems that the most complete and accurate results might probably be achieved by a combination of automated and manual annotation. First, the automated process could 'suggest' annotations, which later can be revised by the users. Furthermore, the general vs. specific problem could be improved by recognizing the candidate lists in the autosuggestion by means of grouping specific items below general items. To improve the wrong entity rate the differences between entities should be made more clear, e.g. with sophisticated entity summaries.

## 5.2 Navigation and Exploration Interfaces

The preliminary user study on the navigation and exploration interfaces resulted in further insights on how Linked Data based visualizations should be presented to the users. The *Infobox* visualization was preferred the most by the participants. During the navigation task, users had no problems to find relevant information and commented positively on the way the additional information is presented. All but one participant could imagine to use the *Infobox* visualization on the Web. About 60% of the participants could imagine to use the *Relation Browser* as well. However, some users had difficulties understanding the interface. Especially the direction of line connectors was not intuitive for most lay-users. Also many users had problems finding the hidden entities in the grid-view via the pagination bars or the +icon on the bottom of an entity tile. Further comments on the *Relation Browser* revealed that by activating the *Relation Browser*, the context of the blog-post goes missing. The *Recommendation* visualization was used quite intuitively by the participants. Most users could imagine using the visualization, even though not all did understand that the recommendations are based on the specific focus-entity instead of the current blog-post as a whole.

## 6   Conclusion

In this paper, we have proposed two semantic annotation interfaces to enable authors to link their content with DBpedia entities as convenient, fast, and accurate as possible and Linked Data based visualizations to enable users to actively explore and navigate the entire content of a platform. The qualitative user study on both annotation interfaces resulted in the insight that the *Inline Annotator* delivered better usability results due to its compactness, faster interactions, and the possibility to better retain the context of the text. The *Modal Annotator* however enabled the users to annotate more accurately. In future research it will be investigated how to combine the best features of the modal and inline interface to enable better manual annotation and how automated and manual annotation tools can be combined to achieve the best results. The preliminary qualitative user-study on the *Infobox*, the *Relation Browser*, and the *Recommender* revealed that the *Infobox* was highly preferred by the participants and well understood while many improvements have to be made on the *Relation Browser* and *Recommender* to enable better exploration and navigation. Further research regarding the visualization tools will focus on how to display the additional information without hiding the context of the blog-post and how to enable the user to better navigate through entities which are not displayed in the first overview.

## References

1. A.-S. Dadzie and M. Rowe. Approaches to visualising linked data: A survey. *Semantic Web Journal*, 2(2):89–124, 2011.
2. T. Di Noia and V. C. Ostuni. Recommender systems and linked open data. In *Reasoning Web. Web Logic Rules. 11th International Summer School 2015*, Lecture Notes in Computer Science, pages 88–113. Springer, 2015.
3. A. Khalili and S. Auer. WYSIWYM – integrated visualization, exploration and authoring of semantically enriched unstructured content. *Semantic Web Journal*, 6(2):259–275, 2014.
4. A. Khalili, S. Auer, and D. Hladky. The RDFa content editor - from WYSIWYG to WYSIWYM. In *Proceedings of of COMPSAC 2012 - Trustworthy Software Systems for the Digital Society*, 2012.
5. A. Khalili, S. Auer, and A.-C. N. Ngomo. context – lightweight text analytics using linked data. In *11th Extended Semantic Web Conference (ESWC)*, pages 628–643. Springer, 2014.
6. C. Morbidoni and A. Piccioli. *The Semantic Web: ESWC 2015 Satellite Events*, chapter Curating a Document Collection via Crowdsourcing with Pundit 2.0, pages 102–106. Springer, 2015.
7. J. Osterhoff, J. Waitelonis, and H. Sack. Widen the Peepholes! Entity-Based Auto-Suggestion as a rich and yet immediate Starting Point for Exploratory Search. In *Proceedings of 2nd Workshop Interaction and Visualization in the Web of Data (IVDW)*. GI, 2012.

8. T. Tietz, J. Waitelonis, J. Jäger, and H. Sack. Smart media navigator: Visualizing recommendations based on linked data. In *Proceedings of the Industry Track at the International Semantic Web Conference (ISWC)*, pages 48–51, 2014.

9. T. Trinh, P. Wetz, B. Do, P. R. Aryan, E. Kiesling, and A. M. Tjoa. An auto-complete input box for semantic annotation on the web. In *Proceedings of the International Workshop on Visualizations and User Interfaces for Ontologies and Linked Data co-located with 14th International Semantic Web Conference (ISWC)*, pages 97–103. CEUR-WS, Vol. 1456, 2015.

10. R. Usbeck et al. Gerbil: General entity annotator benchmarking framework. In *Proceedings of the 24th International Conference on World Wide Web (WWW)*, WWW '15, pages 1133–1143. ACM, 2015.

11. J. Waitelonis, C. Exeler, and H. Sack. Linked Data Enabled Generalized Vector Space Model to Improve Document Retrieval. In *Proceedings of NLP & DBpedia 2015 workshop in conjunction with 14th International Semantic Web Conference (ISWC)*. CEUR-WS, Vol. 1486, 2015.

12. J. Waitelonis and H. Sack. Named entity linking in #tweets with kea. In *Proceedings of 6th workshop on 'Making Sense of Microposts', Named Entity Recognition and Linking (NEEL) Challenge in conjunction with 25th International World Wide Web Conference (WWW)*, 2016.