computing@computingonline.net
www.computingonline.net

Print ISSN 1727-6209
On-line ISSN 2312-5381
International Journal of Computing

INTERACTIVE KNOWLEDGE VISUALIZATION TOOLS FOR EXHIBITION CURATION

Jing He, Joachim Quantz

ART+COM AG, Kleiststr. 23-27, 10787 Berlin, Germany, www.artcom.de
Jing.He@artcom.de, Joachim.Quantz@artcom.de

Paper history:
Received 03 December 2017
Received in revised form 25 July 2018
Accepted 28 August 2018
Available online 30 September 2018

Keywords:
Digital Curation;
Knowledge Visualization;
Information Extraction;
Usability;
User Experience;
Knowledge Workers.

Abstract: This paper presents interactive knowledge visualization tools supporting knowledge workers in the process of curating digital content for exhibitions, showrooms, visitor centers or museums. The tools developed in the research project DKT (Digital Curation Technologies), funded by the Federal Ministry of Education and Research (BMBF), use language and knowledge technologies (such as information extraction, image recognition, classification and clustering) to automatically process digital multimedia content and then provide interactive visualizations of the results. The tools are thus not meant to replace knowledge workers but rather to support them and allow them to handle more content in a shorter span of time while maintaining or even increasing the quality of the curation process. Given this particular application scenario, the performance and accuracy of current state-of-the-art algorithms from Artificial Intelligence, though far from being perfect, is already good enough. The focus of the project work presented in this paper is on information extraction and text content.

1. INTRODUCTION

This paper describes work carried out in the research project DKT (Digital Curation Technologies) funded by Germany's Federal Ministry of Education and research [1].

Within DKT, ART+COM developed tools supporting knowledge workers in the process of curating exhibitions. The basic idea is to use language and knowledge technologies (in particular information extraction) to automatically process content (in particular texts) and then provide interactive visualizations of the results. The tools are thus not meant to replace knowledge workers but rather to support them and allow them to handle more content in a shorter span of time while maintaining or even increasing the quality of the curation process.

The paper is structured as follows: Section 2 gives a short introduction into exhibition curation, providing the background and the state of the art for the tool development described in this paper. Section 3 then presents the concept of digital curation tools, both on a generic level and with respect to the specific requirements arising in exhibition curation. The tools themselves are described in detail in Section 4 (Information Extraction) and Section 5 (Knowledge Visualization). This includes presentation of the general approach and its underlying rational, details regarding the implementation, and examples and screen shots for illustration. Finally, Section 6 contains a brief summary and an outlook towards future work.

2. CURATING EXHIBHITIONS

In order to better understand the motivation underlying the tool development described in this paper, this section gives a brief description of the general background and the state of the art in exhibition curation, as performed by knowledge workers at ART+COM.

Examples for media-rich exhibition design in cultural and commercial sectors include a zoo for micro-organisms (Micropia) in Amsterdam, an experience centre on Viking history in Denmark or a Product Info Center for BMW in Munich [2].

For every new exhibition, the life cycle of the

curation process starts with in-house knowledge workers quickly familiarizing themselves with the exhibition's topic, usually by sighting briefing material offered by clients as well as by learning about the general context and related aspects of the respective domains, e.g. by reading books or accessing information freely available on the Internet.

In a second phase, relevant aspects of the topic are identified and grouped into subtopics, e.g. by assigning them to specific rooms or spaces within the exhibition or to specific exhibits.

The work described in the following focuses mostly on support for the initial research phase. However, work in the project DKT also aimed at covering the whole life cycle of exhibition curation.

The knowledge workers at ART+COM absorb and structure information in various forms and transform it into an enriched and contextualized kind of knowledge on the base of the processed material.

Their work is always curatorial in the wider sense of curation as transforming data of all kinds into content, and it is often curatorial in the traditional sense: developing ideas for exhibitions or visitors' centres that will then be expressed by media enriched exhibits.

Both types of curatorial activity are complex processes. They involve routine tasks like the fast sighting, highlighting and sorting of material, and they require a creative re-combination of material and thus decision-making or the inclusion of already acquired knowledge or other data.

Currently, tools like eMail, Excel, Word, etc. are used for collecting and processing material – without any support for automation. The tools provided by language and knowledge technology, on the other hand, are not yet usable by knowledge workers in their everyday routine.

3. DIGITAL CURATION TOOLS

3.1 AUTOMATIC CONTENT ANALYSIS

The first step in the development of digital curation tools consists in the selection and integration of appropriate solutions to automatically analyze digital content.

To do so, a wide range of solutions from Artificial Intelligence (AI) is available, such as information extraction [3] (e.g. identifying names and noun phrases in a text document), image recognition [4] (e.g. detecting persons or objects in a picture), classification [5] (e.g. generating metadata on artistic styles for a painting), or clustering [6] (e.g. grouping images with similar content).

The algorithms currently available differ considerably with respect to accuracy, robustness, or scalability – depending, for example, on the type of content (e.g. text, speech audio, image, or video), the domain of the content (e.g. general news, science, culture, poetry), or the amount of training data available.

While the development of fully automatic tools is thus still a major challenge, the results are often good, robust, and fast enough to support knowledge workers in their everyday activities – speeding up their work while also improving its quality.

3.2 SEMI-AUTOMATIC TOOLS

The semi-automatic tools developed by ART+COM in the project DKT use intelligent algorithms as an internal basis and embed them into interactive tools for knowledge workers.

The focus is on maximizing usability and user experience for the knowledge workers using the tools. No attempts have been made to improve the performance of the integrated algorithms themselves.

The rationale underlying this approach is twofold: On the one hand, currently available algorithms produce results, which are good enough to assist knowledge workers in their curation activities; on the other hand, these algorithms lack sophisticated user interfaces to make them readily usable for knowledge workers.

It should be noted that this is not meant as a criticism against researchers in Artificial Intelligence, but rather as a suggestion for a division of labor between research on basic AI algorithms and user-centered design of smart tools. The former should focus on improving accuracy, scalability, robustness, etc., whereas the latter should focus on user experience and usability, taking into account the capabilities and limitations of the respective algorithms integrated in the tools.

Take as an example information extraction, which will be presented in more detail in the next section. If a knowledge worker needs more time to understand the output of an intelligent text analysis algorithm than to read the text herself, there is no value add.

The main purpose of the tools presented in this paper is therefore to integrate intelligent algorithms into usable tools providing knowledge workers with

- easy-to-grasp overviews,
- extensive drill-down functionality,
- support of the entire curation life cycle;
- seamless integration with IT environments.

Moreover, the tools were also designed to take into account current limitations with respect to accuracy and issues regarding performance and scalability. Knowledge workers can thus add results not found by the algorithms or delete "false" results. And results are presented incrementally, eliminating

the need to wait for the algorithms to finish completely.

More details on the involvement of knowledge workers in the design and evaluation process can be found in [7, 8].

4. INFORMATION EXTRACTION

The information extraction implemented in ART+COM's DKT prototype, a web application using RESTful APIs, comprises three steps:

- Recognizing named entities in texts
- Mapping them to Wikidata entries
- Selecting relevant properties/relations

These steps are described in detail in the section below.

4.1 NAMED ENTITY RECOGNITION

Knowledge workers can trigger information extraction by importing existing documents, e.g. briefing materials provided by the client, or by performing an explorative web search with keywords.

In either case, the content is automatically submitted to a service performing Named Entity Recognition (NER) provided by the DFKI (German Research Center for Artificial Intelligence), an academic partner in the DKT project. NER extracts information about people, places, and organizations from arbitrary text documents [9].

The service also supports extraction of domain-specific entities, but needs to be trained with respective training corpora of sample texts to be able to do so. Given the application scenario described above, compiling such corpora is not realistic and therefore the generic NER service was used.

For instance, the historical text about the Vikings that was used in the knowledge workers' research for the Viking Experience Center, when fed through the NER service, generates a list of entities such as king "Gorm the Old," king "Harald Bluetooth," "Sweden," "Denmark," etc.

There are a couple of challenges for NER services: on the one hand, natural language is inherently and notoriously ambiguous, i.e. depending on the context, a phrase can have very different meanings or refer to different entities [10]. On the other hand, natural language often employs so-called anaphora, e.g. pronouns, to refer to previously mentioned entities, requiring complex anaphora resolution [11].

The following section describes an approach towards disambiguation based on computing semantic similarity from Wikidata information. It also mentions term shadowing and term construction as comparatively simple techniques for anaphora resolution. However, complex anaphora resolution (e.g. computing the year referred to by the phrase "one year later") should be incorporated into the NER service itself and is beyond the scope of this paper.

4.2 WIKIDATA GRAPH SERVICE

The tools developed by ART+COM in the DKT project use Wikidata [12] as external knowledge base (see the sample entry for Harald Bluetooth in Fig. 1). Thus, all the entities extracted from texts by the NER service, are mapped onto entities in Wikidata, in order to be able to access their properties and relations.

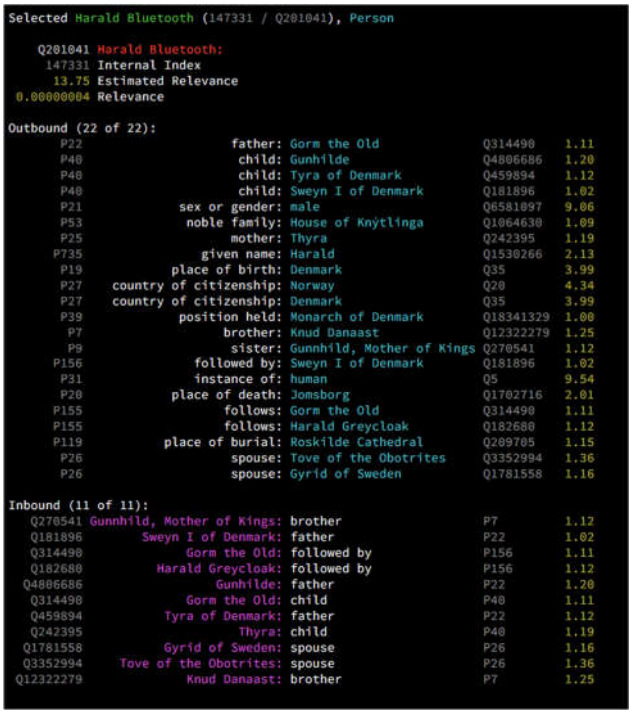

Figure 1 – Wikidata entry for Harald Bluetooth.

One challenge in doing so is the inherent ambiguity of natural language phrases. Thus, there are usually several Wikidata "entity candidates," to which a name or a phrase might refer.

For example, the candidates for the entity "Denmark" are "Kingdom of Denmark," "city in South Carolina," "town in Maine" – possible meanings of "Denmark."

The idea underlying our disambiguation process is to use "semantic similarity" to find the best-suited candidate. A so-called graph service preprocesses the entire Wikidata knowledge base in order to generate a graph that approximates how closely Wikidata entries relate to one another.

This is achieved by using an algorithm from machine learning, such as Word2Vec, that generates a set of vectors for each entity [13]. The semantic

distance between the entities is encoded in these sets of vectors.

When a list of entities from a document is found via the NER service, the graph service performs analysis on the entities and retrieves a set of *entity candidates* for each entity.

The similarity value is computed by an iterative re-sorting process, which compares candidates for each entity to other entities and their respective candidates.

Based on this similarity value the candidate for each entity that best fits in the given context is chosen. "Denmark" resolves to "Kingdom of Denmark" because it relates to king "Gorm the Old" and Harald Bluetooth" better than other suggested candidates in the context of the research on Viking history.

Other techniques addressing entity ambiguity are *term shadowing* and *term construction*. In the text the term "Gorm" and "Gorm the Old" are both mentioned and therefore extracted as separate entities. It is very common that the subject is addressed only with first or last name for the sake of brevity.

Term shadowing resolves these two entities as one. In other cases an extracted entity has incomplete information, i.e. only "Gorm" is mentioned in the text.

Term construction completes the identity of an entity. In the chosen case the system would suggest that "Gorm" means "Gorm the Old."

The following section provides more details regarding the implementation. It should be noted, that the implementation aims at providing a heuristic solution to a complex problem and does not aim at always finding the optimal solution.

Again, a solution which is "good enough" and can be computed in near real-time is entirely appropriate for our application.

4.3 IMPLEMENTATION DETAILS

In order to retrieve the entity candidates efficiently, we have cloned the entire Wikidata graph locally and parsed the required subset of each node's data into an indexed binary format, thereby optimizing both the look-up and bidirectional traversal. This data structure serves as basis for creating the graph where every entity or node in this graph carries both the inbound and outbound connections, in other words, how each entity is connected to others.

The graph service is further refined by Google's page ranking algorithm, which generates a relevance value for each entity [14]. The process iteratively redistributes relevance values along node connections until the system converges to a state where each node's relevance represents how much it is transitively referred to by the network as a whole.

The last stage of pre-processing the graph is building a graph with vectorized nodes via machine learning. The Word2vec algorithm [13] is employed to assign each entity in a vector space with multiple dimensions. Since Word2Vec is designed to be trained on word embeddings from a text, yet our entities are organized in a graph, we needed to find a mapping from one to the other.

The DeepWalk approach [15] is implemented in order to traverse the graph and assemble sequences of entities in a greater range. Not only first-degree connections are considered but also entities connected via multiple nodes in-between, allowing a more accurate semantic representation of a concept by our trained vectors.

The NER service provides a list of words extracted from a corpus of documents. Each word is looked up in the graph service to retrieve all potential matches. Before the retrieval is performed, an NLP technique called Stemming is used to get to trim the incoming word to a common sub-string that is shared among all or most of the words inflections. For instance, "knitted, knitting and knits" are converted to "knit."

After the entity candidates are retrieved, they are then sorted based on a lookup score. This lookup score is partly based on the relevance value derived from the Google page ranking algorithm, and partly based on character casing, i.e., queen vs. Queen, word length and usage as a primary vs. secondary label for an entity, i.e., USA vs. United States of America (see the example for "Harald" in Fig. 2).

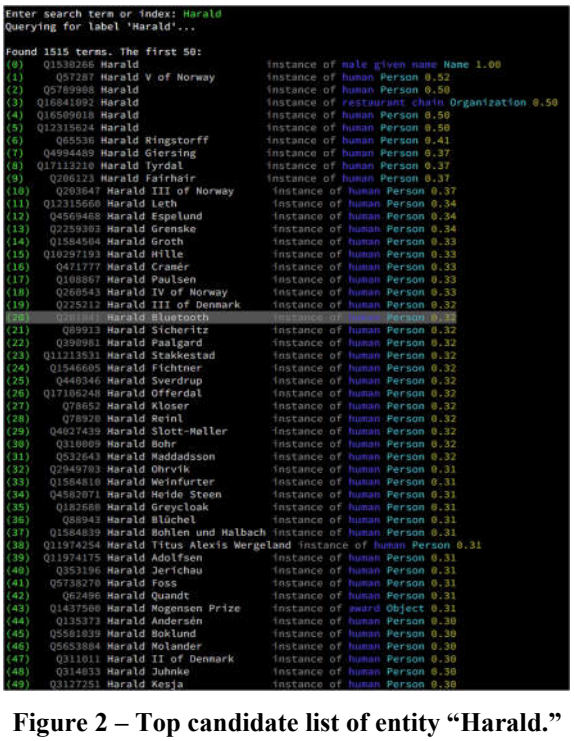

Figure 2 – Top candidate list of entity "Harald."

With a resorted candidate list for each entity at hand, the final step is to resolve the ambiguity of each word extracted from the NLP service and finding the best matching entity candidate to represent its intended meaning. For instance, is "queen" the British rock band or a person at the head of a monarchy? Resolving a word's ambiguity involves an iterative cycle of grouping and regrouping different sets of entity candidates around each entity label in order to progressively rate the entity candidates based on its intended meaning. A group vector is produced each time based on the similarity value amongst the entity candidates. Eventually the iteration ends when the best matching candidate for an entity label is found (see Fig. 3).

Figure 3 – Iterations of resolving entity labels.

A technique called Simulated Annealing [16] is employed to facilitate the creation of groups. By introducing a random factor to the group which over the time decreases its value to "cool down" the randomness, the entity candidates are initially allowed to favor seemingly worse groupings over better ones in order to avoid locking into local minima early on. As the random factor decreases, the top-matching list of candidates becomes more stable until the iteration stops and the ambiguity is resolved.

4.4 PROPERTIES AND RELATIONS

Having determined the most likely candidate in Wikidata, properties and relations are extracted. In doing so we use a very simple top-level ontology, containing top-level concepts such as person, organization, location, etc (see [17] for a detailed analysis of linguistic and cognitive categorization).

The top-level ontology currently comprises 12 concepts, which are hard-wired into the code. Also hard-wired into the code are the mappings from terms used in Wikidata to the terms used in our tools.

For the current purpose, such a hard-wiring is "good enough". However, future work is envisaged to allow a more flexible approach to mapping, supporting different knowledge bases and allowing users to interactively add or change mappings.

The properties and relations are used extensively in the interactive knowledge visualizations presented to the knowledge workers, e.g. to display detailed information, to show connections or to allow filtering and sorting. The next section will provide more details and examples for illustration.

5. KNOWLEDGE VISUALIZATION

In order to present the results of the information extraction described in the previous section to knowledge workers, several visualization methods have been designed and implemented, including a network overview (Fig. 4), semantic clustering (Fig. 5), and timelining and maps (Fig. 6). In addition, the interaction design offers both generic interactions common to all visualizations as well as interactions customized towards the specifics of each respective visualization.

Figure 4 – Network visualization.

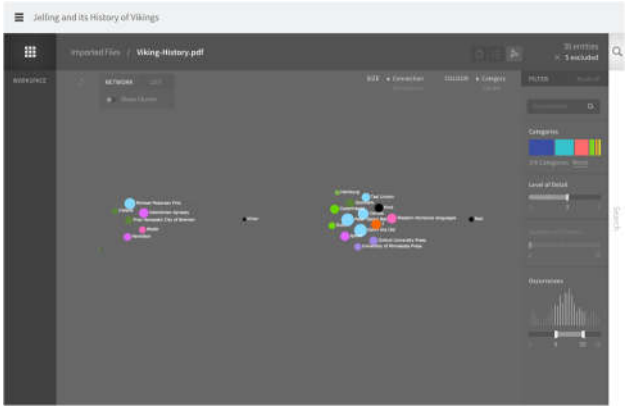

Figure 5 – Semantic cluster visualization.

A visual representation of knowledge structures or ontologies [18] already formed the underlying rationale for Semantic Networks [19] and is also envisaged for the Knowledge Graph [20].

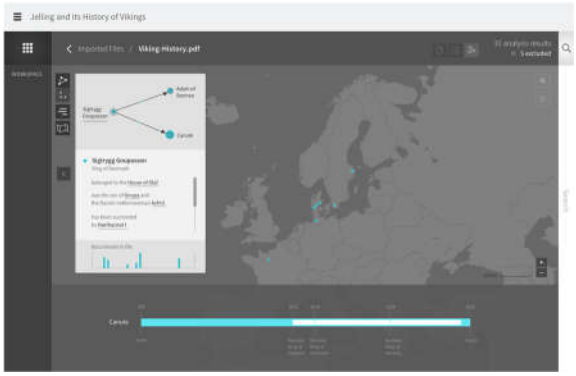

Figure 6 – Timeline and map visualization.

5.1 GRAPH VISUALIZATION

We implemented the visualizations in D3.js [21], a Javascript library optimized for visualizing data with HTML, SVG and CSS. The challenge is to provide intuitive and effective interaction modalities so that the users can gain a quick overview of a specific topic or drill down into the semantic knowledge base to explore deeper patterns or relationships.

While a network (Fig. 4) is effective for exploring semantic relationships amongst extracted entities, semantic clustering (Fig. 5) provides a good overview of entities closely related to each other in groups.

Timelines and maps (Fig. 6), in addition, offer a geo-temporal overview of extracted entities. Combining all of the above-mentioned visualisations for doing research on "Harald Bluetooth," for example, the user can explore the places and people to which he is connected in the network visualization, view his timespan of reign and family lineage along the timeline, and learn about notable events that are highlighted on the map.

5.2 INTERACTION DESIGN

In terms of interaction modalities, the user can directly manipulate the graph, e.g. to zoom and pan. Moving the cursor above entities triggers a highlighting, which displays entity properties found on Wikidata. The user can also select and focus on the connections between two entities in the *detailed connection visualization* to investigate how they are related (Fig. 7). Harald Bluetooth's connection to Gorm the Old, for instance, is encoded with several connection possibilities. Harald is the son of Gorm and his heir to the throne.

Besides the direct interaction with the visualization, we also implemented interactive filters allowing the user to effectively gain specific information as needed. For instance, the category filter allows the user to filter entities that are only "person" or a combination of "person," "location," or other category labels that appear relevant to the user. In the future, the user can also add new category labels, filter by number of occurrence, or by connection types.

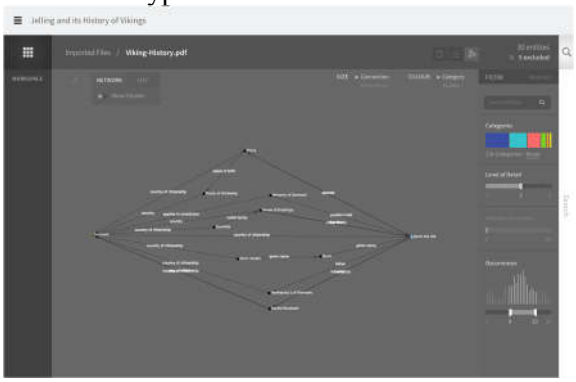

Figure 7 – Detailed connection visualization.

The prototype is designed with our in-house knowledge workers' iterative work cycle in mind. Since they often deal with multiple projects at the same time, the interface allows the users to switch between projects at any time during their research process while the research materials, whether imported by the users or search results from the web, are saved within the related project. This way the users can continue from where they have left off in the previous research session.

The design template for each project's home screen consists of two main areas. On the left side the user can import multiple text files or view imported files, while on the right side the user can start an explorative search on the web with advanced search options such as knowledge sources, depth of search, and amount of returned results (Fig. 8).

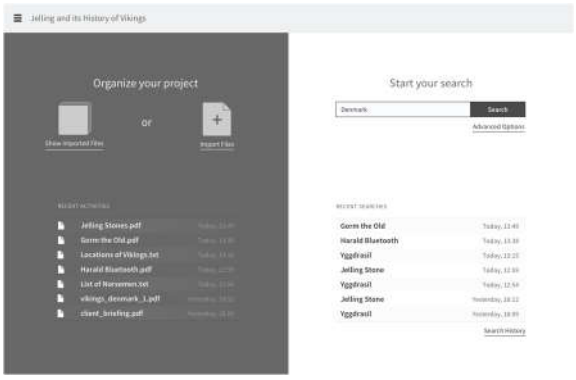

Figure 8 – Project home screen.

Following the *import file* or *open collections* option, the interface displays the *collection view* and shows imported files. In the future we will develop the option to create new collections by the user, i.e. by topics much like using folder structures.

Within each project space there are six main interface areas:

- The current project title and menu dropdown to switch to other projects are located in the top bar.
- On the left side, a collection icon offers a shortcut for the user to quickly navigate back to the top level collection view. An empty space labeled "workspace" where the users can add shortcuts to collections or documents allows the users to gain quick access to research materials they are currently working on or save their findings from search or imported documents.
- The *search bar* on the far right is where the users can jump over to the search results. For example, the user finds "Gorm the Old" an interesting entity in the imported document, she could start a search from the entity to learn more about it in depth.
- The *header bar* area contains main content navigation structure. A breadcrumb trail displays where the user is located within the project. The three main document-centric views are accessible at all times so that the user can freely toggle amongst: *document view*, *entity list view*, and *entity visualization view*.
- Within each content view an *interaction panel* is always present on the right-hand side where context-specific interaction options are displayed. Although interaction options change based on the different content views, the functionality of the area remains consistent so that the user can become familiar with different interaction options faster.
- In the *document view* (Fig. 9), the user can browse through the original text with extracted entities highlighted. In the *interaction panel* area a search field offers search options on any given entities.

The *entity list view* (Fig. 10) consists of a list of entities which can be sorted by number of occurrences in the text, category types or alphabetic order.

A short entity statement accompanies each entity, i.e. "Gorm the Old was the king of Denmark." In the *interaction panel,* a search bar to search for existing entities is available as well as a category filter to filter entities by category types.

The third content view is the *entity visualization view* (Fig. 11).

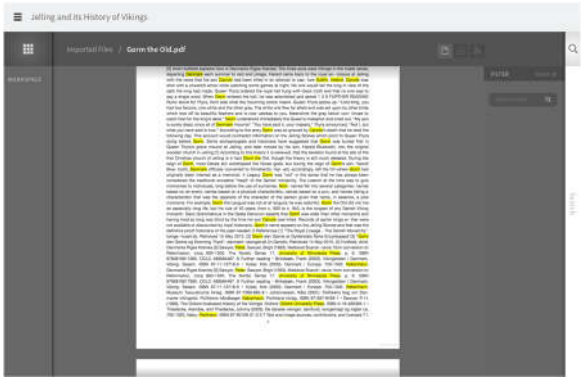

Figure 9 – Document view.

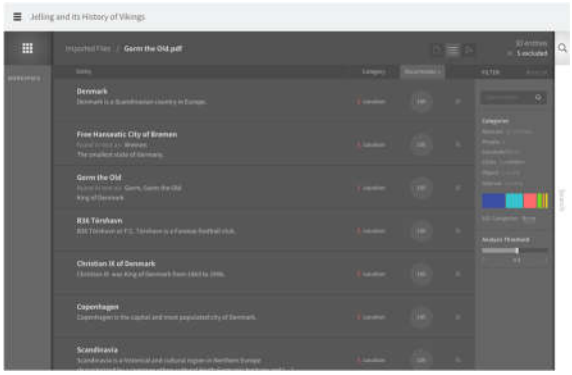

Figure 10 – Entity list view.

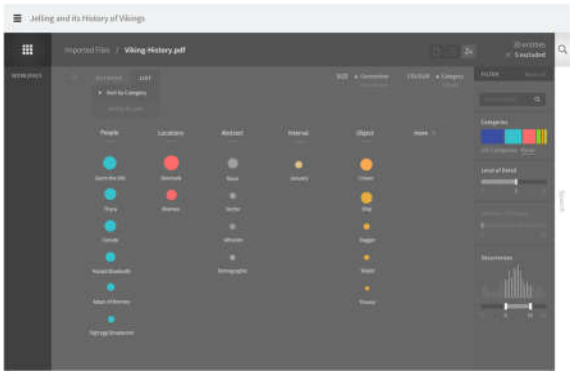

Figure 11 – Entity visualization view.

6. CONCLUSION

We have presented interactive knowledge visualization tools supporting knowledge workers in the process of curating digital content for exhibitions. Some of the results are also applicable to other usage scenarios.

The work has been part of the research project DKT (Digital Curation Technologies), which overall demonstrated the usefulness of language and knowledge technologies in the context of digital content curation.

Currently available intelligent algorithms for content analysis are already accurate, robust and fast enough to assist knowledge workers in everyday tasks – provided they are enhanced by user interfaces and interaction design focusing on usability and user experience.

Future work will focus on true multimedia support (e.g. information analysis for audio, image, video in addition to text) as well as re-using techniques in building interfaces for "smart exhibits." In the latter case the users will be visitors of an exhibition and not just knowledge workers – leading to slightly different requirements regarding usability and user experience.

7. REFERENCES

[1] http://digitale-kuratierung.de

[2] https://www.artcom.de

[3] J. Piskorski, R. Yangarber, "Information extraction: past, present and future," in T. Poibeau et al. (eds.), *Multi-source, Multilingual Information Extraction and Summarization 11,* Springer-Verlag Berlin Heidelberg, 2013, pp. 23-49.

[4] *TensorFlow: Image Recognition, Tutorial*, https://www.tensorflow.org/tutorials/image_recognition.

[5] L. Pierson, *Classification Algorithms Used in Data Science*, http://www.dummies.com/programming/big-data/data-science/classification-algorithms-used-in-data-science/.

[6] G. Seif, *The 5 Clustering Algorithms Data Scientists Need to Know*, https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68

[7] G. Rehm, J. He, J. Moreno-Schneider, J. Nehring, J. Quantz, "Designing user interfaces for curation technologies," in *19th International Conference on Human-Computer Interaction – HCI International 2017*. Vancouver, Canada, July, 2017.

[8] J. He, N. Göhlsdorf, "Digital Curation Tool in the Age of Semantic Technology," in C. Busch, C. Kassung, J. Sieck (Eds.), *Kultur und Informatik: Mixed Reality 2017*, pp. 193–208.

[9] P. Bourgonje, J. Moreno-Schneider, G. Rehm, F. Sasaki, "Processing document collections to automatically extract linked data: semantic storytelling technologies for smart curation workflows," in A. Gangemi, C. Gardent, (Eds.), *Proceedings of the 2nd International Workshop on Natural Language Generation and the Semantic Web (WebNLG 2016)*, pp. 13–16.

[10] Anjali et al., "Ambiguities in Natural Language Processing," in *International Journal of Innovative Research in Computer and Communication Engineering*, Vol. 2, Special Issue 4, pp. 392-394, 2014.

[11] Poesio et al., *Anaphora Resolution: Algorithms, Resources, and Applications*, Springer, 2016.

[12] https://www.wikidata.org/wiki/Wikidata:Main_Page

[13] Q. Le, T. Mikolov, "Distributed Representations of Sentences and Documents," https://cs.stanford.edu/~quocle/paragraph_vector.pdf.

[14] R. S. Wills, "Google's PageRank: The Math Behind the Search Engine," May, 2006, http://www.cems.uvm.edu/~tlakoba/AppliedUGMath/other_Google/Wills.pdf.

[15] B. Perozzi, R. Al-Rfou, S. Skiena, "DeepWalk: Online Learning of Social Representations," June 2014, https://arxiv.org/pdf/1403.6652v2.pdf.

[16] D. Bertsimas, J. Tsitsiklis, "Simulated Annealing," *Statistical Science,* Vol. 8, Issue 1, pp. 10-15, 1993, http://web.mit.edu/jnt/www/Papers/J045-93-ber-anneal.pdf.

[17] G. Lakoff, *Women, Fire and Dangerous Things*, University of Chicago Press, 1990.

[18] A. Katifori et al. "Ontology visualization methods – a survey," http://dit.unitn.it/~p2p/RelatedWork/Matching/a10-katifori.pdf.

[19] J. F. Sowa, "Semantic Networks," http://www.jfsowa.com/pubs/semnet.htm.

[20] Google: The Knowledge Graph, https://www.google.com/intl/es419/insidesearch/features/search/knowledge.html.

[21] D3.js: Data-Driven Documents, https://d3js.org/.

Jing He *holds a BA in Fine Arts (1998–2002) from Charleton College, Minnesota, USA and an MFA Design and Technology (2002–2004) from Parsons School of Design, New York, USA. She joined ART+COM in 2004 where she currently holds the position of Design Research Lead. Her major expertise and interests include interaction design, human-centered design as well as solutions for health and life sciences.*

Joachim Quantz *has studied Computer Science (Diploma 1988, PhD 1995) and Linguistics/Philosophy (M.A. 1992), at Technische Universität Berlin. He has joined ART+COM as Head of Research in 2008. His areas of expertise include Human Computer Interaction, Semantic Technologies, Smart Home, Mixed Reality, Business Process Management and Mobile Apps.*