AUI4RFQ: How Can Adaptive User Interfaces Support Document-Centric Knowledge Work?

Rene Kaiser
rkaiser@know-center.at
Know-Center – Research Center for Data-Driven Business & Big Data Analytics
Graz, Austria

ABSTRACT

Request for quotation (RFQ) is a process that typically requires a company to inspect specification documents shared by a potential customer. In order to create an offer, requirements need to be extracted from the specifications. In a collaborative research project, we investigate methods to support the document-centric knowledge work offer engineers conduct when processing RFQs, and started to develop a software tool including artificial/assistive intelligence features, several of which are based on natural language processing (NLP). Based on our concrete application case, we have identified three aspects towards which intelligent, adaptive user interfaces may contribute: adaptation to specific workflow approaches, adaptation to user-specific annotation behaviour with respect to the automatic provision of suggestions, and support for the user to maintain concentration while conducting an everyday routine task. In a preliminary, conceptual research phase, we seek to discuss these ideas and develop them further.

KEYWORDS

HCI; NLP; AI; adaptive user interfaces; knowledge work; document-centric; concentration work; RFQ; tender;

1 INTRODUCTION

Many companies in the business-to-business (B2B) sector are providing not mass products but custom goods, services, technology or other products to client companies. In what is called a *request for quotation* (RFQ) process, a customer typically contacts them with a detailed specification of the desired product. The company then needs to inspect the specifications and extract all the requirements contained in them. It assesses whether it is able to fulfil these requirements, if necessary clarifying with the customer whatever information is ambiguous or missing. Based on the requirements extracted, some design decisions may need to be taken in order to decide what exactly is offered to the client. Ultimately, the company compiles an offer for the customer, including a detailed description of the offer, and a price that is usually binding.

The company providing the offer may be facing the risk of missing details in the specifications and as a result offering for a price that is not profitable. At the same time, these companies try to limit the amount of work that goes into creating offers, and can't afford to strive for perfection.

In the realm of a collaborative research project with an industry partner which started in 01/2019, we are aiming at supporting such RFQ processes with a socio-technical intervention, which means we are both i) investigating ways to improve work processes, and are ii) designing and developing a software tool to support document-centric knowledge work. The project is combining two research focus areas: a HCI perspective to design the tool and work processes, and a NLP part to integrate intelligent features.

Several algorithms in the NLP processing chain (e.g. search, classification) shall apply what can be called *AI* capabilities – artificial or assistive intelligence, depending on the definition of the term, and also the complexity and maturity level that will increase during the course of the research project. The basic approach is to analyze specification documents and to collect usage and document annotation data when the tool is being used. This data is used to iteratively (re-)train and improve machine learning models to support the user to process the current RFQ case. Historic data from previous RFQ cases shall be exploited to speed up and improve the quality of the current task.

With respect to the HCI part of this endeavour which is most relevant for the AI4AUI workshop, the central research question which we aim to reflect is: given what our tool knows about the current RFQ case, an already rich and growing database of historic cases, and the specific work and annotation behaviour of the current user, in what ways can an intelligent, adaptive UI best support the user to create the current offer? At the overlap of our HCI and NLP aspects, how can we make use of the data which we are already processing to also adapt the user interface (UI) of the tool itself, such that it supports the user best while at the same time not confusing the user with unexpected self-adapting behaviour?

Our research around this topic connects to previous work on supporting concentration work in control centers, see [6] and [13]. It builds as well on background on the auto-adaptive presentation of personalized live video content (cf. [7]).

While research has looked at document-centric knowledge work in general, and published findings we can build on (e.g., [8], [5], [11]), research on technology supporting the specific task of RFQ processing is scarce. This is surprising given the fact that this is a very relevant process and cost factor for many companies. We are not aware of any prior research literature on how to design and apply adaptive user interfaces in the specific context of RFQ processing.

IUI '20 Workshops, March 17, 2020, Cagliari, Italy

2 SPECIFIC CASE: THE ISA PROJECT

In principle, there are very different kinds of RFQs processes. The following describes some of the specific aspects of our collaborative project called *ISA*. Our insights are stemming mainly from interviews with offer engineers which we conducted based on the *contextual inquiry* method (see [4]).

The industrial partner in our research project is producing machines worth several millions of Euros. Offer engineers process RFQ case after case. As far as possible, they avoid processing multiple cases in parallel. Depending on the complexity of the product type and other factors, the time to process a case is ranging from less than a working day on the low end, to about 2 weeks as the average case, to around 2 months as a practical upper limit. Specifications shared by their customers typically involve multiple documents and sum up to several hundreds of pages. Some segments of these specification documents are more relevant than others (e.g., norm specifications or annexes with details that are not relevant for the RFQ process).

Our target users are offer engineers who work in open plan offices and are hence dealing with certain distractions in their working environment. Most of them use two large screens on their working desk. We decided to limit ourselves to a graphical user interface only, to not use other modalities like for example the vibration feedback common with mobile phones. Researchers have, however, suggested numerous other methods including document annotation based on eye gaze, see [2]. Specifically, we do not consider audio as input or output channel of the tool, due to the open plan office context on the one hand, and the fact that wearing headphones all day is not bearable on the other.

Offer engineers are conducting a knowledge-intensive routine task consisting of the following compound steps:

(1) Extract information from specifications, whatever is relevant for the preliminary design of the product or cost-relevant. Concentrated reading, counter checking, combining individual bits of information.
(2) Asynchronous communication with customers for clarifications.
(3) Design of the product based on the requirements extracted. Calculation of a price, compilation of the offer documents.

These offer engineers we seek to support need to skim and carefully read hundreds of pages of text every day, in what we consider highly concentrated knowledge work. We consider it very challenging to maintain concentration and keep a big picture overview over several hours of reading. With respect to the requirements that offer engineers extract from the specification texts, we distinguish two top-level cases:

(1) There is a closed set of known product parameters. Based on information found in the specifications, the offer engineer needs to derive how to parametrize the product design.
(2) Requirements of any kind may appear in the specifications. There is no underlying blueprint solution which is customized.

In the ISA project, we are facing case 1). The ISA tool will be a Web application, hence we are constrained by with certain practical UI limitations. We are dealing with professional expert users.

3 WORKSHOP DISCUSSION ASPECTS

We have identified three distinct aspects where we hypothesize that certain capabilities of an adaptive user interface can support the tasks of offer engineers. With respect to all three of these ideas, we are in a preliminary, experimental research phase.

3.1 Adapting the UI to workflow approaches

Based on an analysis of current work practices and several design cycles (cf. Contextual Design method [4]), we concluded that offer engineers could conduct their work based on very different workflow approaches, each approach with certain strengths and downsides. We work on the assumption that a tool supporting only a single workflow approach is not the best solution, even though there would as well be advantages to unifying all users' work procedures very strictly. Besides many other options, these are three examples of such workflows:

(1) Reading specifications page by page and extracting whatever relevant information is found. Support this process with specific NLP features that can be summarized as 'reading aid', for example pre-highlighting what the tool considers relevant, to steer the attention of the offer engineer. The UI could also suggest on the level of text segments when to skim superficially, and when to read exactly.
(2) Iterating through a list of known product parameters and searching for specification sections containing relevant information, by executing search patterns and looking at ranked lists of automatic suggestions.
(3) A two-step process: first, iterate over specification documents to annotate whatever is considered to have relevance for certain product parameters. In a second step, taken into account all annotations made, decide on the final parametrization value.

The ISA tool shall support several complex workflows and reasonable switching from one to another. We not only hypothesize this flexibility to increase the user acceptance of the tool, but also that it leads to better quality results when catering for the individual preferences and strengths of individual offer engineers.

The introduction of a software tool that offer engineers would start to use for most of their working time is a considerable intervention full of pitfalls: the offer engineers' experience and domain knowledge is a crucial asset for our partner company. Some of these target users are more open to adapt to a new tool than others. The affinity to use digital tools may be very different among offer engineers that are between 20 and 60 years of age, for example. Offer engineers may face the introduction of an AI system with resistance, fearing that their work is about to be replaced. The more experienced of them may have optimized their work practices over many years and have a hard time changing. Could recent research advances in the realm of adaptive user interfaces help mitigate these challenges?

Based on significant amounts of historic data, we shall be able to derive detailed user profiles, to the extent that privacy and employee rights allow to do so. Such profiles can be collected or learned, and later exploited for the sake of adapting workflows and the corresponding UIs. Due to the multitude of options and potential downsides such as a redundancy of UI elements, we consider an adaptive UI approach to be suitable in principle, and therefore seek to discuss this aspect at the AI4AUI workshop. Users may prefer to dissect their tasks differently, may prefer to process certain aspects in different orders, or use different overarching workflows and search methods to search for certain kinds of product parameters. However, the user shall in any case retain control over how automatic the tool can adapt the workflow.

3.2 Adaptation to annotation behaviour

Another aspect which may be supported by adaptive UI functionality is to adapt to very specific user behaviour, in our case the annotation behaviour of the users on the one hand, and the automatic support of the system on the other hand. We assume a benefit of adapting the latter to provide personalized suggestions, based on what the tool learns about the user through the former aspect. The framework of our system puts us in place to study human routine work behaviour by capturing fine-granular user behaviour trace data (cf. [1]).

Suggestions can be filtered and ranked differently based on what a user is expected to utilize, text spans of suggestions can be more or less narrow (balancing focus vs. context), can show or hide certain metadata predicting its usefulness, etc. Any adaptation shall be specific to the user, the type of product, the language of the RFQ case (auto-translate to adapt to current offer), etc. Another example is to adapt the suggestions based on the remaining timeframe to complete an offer, essentially helping the user to balance the trade-off to look at the specifications in great detail, or to invest less time and accept greater risk.

As an underlying concern, users may regard their annotation tasks as an overhead. The immediate benefit of annotation actions is unapparent. It comes later, indirectly, since annotation data is mainly used by NLP modules using machine learning techniques. As a hypothetical worst case scenario, users may oppose the annotation tasks in fear their job to become obsolete once the AI can automate their human labour.

For the initial roll-out of the tool to a small set of test users, the intelligent NLP features will be rather limited.Only later, once a significant amount of annotation data has been collected, the NLP features will start to provide useful and accurate results, providing an immediate reward for the user, presumably increasing the motivation to annotate. At least we can hypothesize that the users will perceive it that way and later evaluate to what degree this assumption holds. To this end, we would like to reflect the question how an adaptive UI can support user motivation in such contexts. Could the deliberate use of ambiguity encourage personal engagement with a system [3]?

Another question in that regard is weather in our specific case there is a relevant downside to providing intelligent suggestions which are often but not always reasonable. What may happen is that users trust the tool too much, failing to realize when it doesn't work. The result may be a decrease in the quality of the offer engineers' work, apart from the positive effect that the required time to process an RFQ is decreasing at the same time. Overall, trust is an important human factor which we want to design for in our socio-technical intervention.

3.3 Supporting concentration work

RFQ processing in our project case is a routine everyday task. Specialized offer engineers often process hundreds of document pages within a single working day. A great amount of concentration is required to do that without mistakes and without missing key information contained in the documents. The processing of RFQs in other words is a routine task and a task requiring highly skilled knowledge work at the same time.

Our research question in that regard is how intelligent, adaptive UI functionality could support the users to manage and maintain concentration levels. Since we are addressing a routine task, the goal is not to achieve as much as possible over a single working day, but to design means of support that are sustainable long-term without causing any harm to the users.

Since the ISA tool is intended to be used over long periods of time, basically most of the offer engineers' working day, there is scope for the tool to continuously monitor user behaviour, infer whether any support is currently suitable, and adapt the UI in whatever form to do so. One step towards such a solution is to analyse user interaction data to detect the current concentration state of the user, comparable to how other researchers tried to detect boredom in different contexts before [10]. Should the tool not be able to acquire enough data to build specific models (cf. [9]) and detect relevant states, it could also prompt the users for input and reflection in a learning phase training AI models.

Understanding what exact adaptation behaviour actually supports concentration is up for research and evaluation. We have discussed only preliminary ideas with target users in interviews.

One aspect may be to avoid distraction, interruptions and activity switches, and also to minimize multitasking (cf. [12]). Concentration work may be scheduled in intervals such that the tool is encouraging users to get regular breaks, suggesting user-specific working rhythms. The tool could schedule tasks, assuming for example better concentration ability in the mornings rather than right after lunch breaks. Could an adaptive UI which actively creates variety and alternation in a sequence of work tasks support users, or should it rather strive for the opposite and avoid variety wherever possible to support concentration? A particular aspect in that regard is how to sequence and mix the tasks of (i) reading specifications and (ii) communicating with customers for clarifications.

ACKNOWLEDGMENTS

The Know-Center is funded within the Austrian COMET Program – Competence Centers for Excellent Technologies – under the auspices of the Austrian Federal Ministry of Transport, Innovation and Technology, the Austrian Federal Ministry of Economy, Family and Youth and by the State of Styria. COMET is managed by the Austrian Research Promotion Agency FFG.

REFERENCES

[1] Nikola Banovic. 2018. *Computational Method for Understanding Complex Human Routine Behaviors.* Ph.D. Dissertation. Carnegie Mellon University.
[2] Georg Buscher, Andreas Dengel, Ludger van Elst, and Florian Mittag. 2008. Generating and Using Gaze-based Document Annotations. In *CHI'08 Extended Abstracts on Human Factors in Computing Systems (CHI EA'08).* ACM, New York, NY, USA, 3045–3050.
[3] William W. Gaver, Jacob Beaver, and Steve Benford. 2003. Ambiguity As a Resource for Design. In *Proceedings of the SIGCHI Conf. on Human Factors in*

Computing Systems (CHI'03). ACM, New York, NY, USA, 233–240.

[4] Karen Holtzblatt and Hugh Beyer. 2016. *Contextual design: Design for life.* Morgan Kaufmann.

[5] Paul Jones, Shivani Sharma, Changsung Moon, and Nagiza F. Samatova. 2017. A Network-Fusion Guided Dashboard Interface for Task-Centric Document Curation. In *Proceedings of the 22nd Int. Conf. on Intelligent User Interfaces (IUI'17)*. ACM, New York, NY, USA, 481–491.

[6] Rene Kaiser and Ferdinand Fuhrmann. 2014. Multimodal Interaction for Future Control Centers: Interaction Concept and Implementation. In *Proceedings of the 2014 Workshop on Roadmapping the Future of Multimodal Interaction Research Including Business Opportunities and Challenges (RFMIR'14)*. ACM, 47–51.

[7] Rene Kaiser and Wolfgang Weiss. 2014. *Media Production, Delivery and Interaction for Platform Independent Systems: Format-Agnostic Media.* Wiley, Chapter Virtual Director, 209–259.

[8] Nishtha Madaan, Hima Karanam, Ankush Gupta, Nitisha Jain, Arun Kumar, and Srikanth Tamilselvam. 2017. Visual Exploration of Unstructured Regulatory Documents. In *Proceedings of the 22nd Int. Conf. on Intelligent User Interfaces Companion (IUI'17 Companion)*. ACM, New York, NY, USA, 129–132.

[9] Antti Oulasvirta. 2019. It's Time to Rediscover HCI Models. *Interactions* 26, 4 (2019), 52–56.

[10] Martin Pielot, Tilman Dingler, Jose San Pedro, and Nuria Oliver. 2015. When Attention is Not Scarce - Detecting Boredom from Mobile Phone Usage. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp'15)*. ACM, New York, NY, USA, 825–836.

[11] Hugo Romat, Nathalie Henry Riche, Ken Hinckley, Bongshin Lee, Caroline Appert, Emmanuel Pietriga, and Christopher Collins. 2019. ActiveInk: (Th)Inking with Data. In *Proceedings of the 2019 CHI Conf. on Human Factors in Computing Systems (CHI'19)*. ACM, New York, NY, USA, Article 42, 13 pages.

[12] Alexander Seeliger, Benedikt Schmidt, Immanuel Schweizer, and Max Mühlhäuser. 2016. What Belongs Together Comes Together: Activity-centric Document Clustering for Information Work. In *Proceedings of the 21st Int. Conf. on Intelligent User Interfaces (IUI'16)*. ACM, New York, NY, USA, 60–70.

[13] Alessandro Vinciarelli, Anna Esposito, Elisabeth Andrér, Francesca Bonin, Mohamed Chetouani, JeffreyF. Cohn, Marco Cristani, Ferdinand Fuhrmann, Elmer Gilmartin, Zakia Hammal, Dirk Heylen, Rene Kaiser, Maria Koutsombogera, Alexandros Potamianos, Steve Renals, Giuseppe Riccardi, and Albert Ali Salah. 2015. Open Challenges in Modelling, Analysis and Synthesis of Human Behaviour in Human-Human and Human-Machine Interactions. *Cognitive Computation* (2015), 1–17.