Model-Based Approach to Design and Develop Usable and Dependable Recommender Systems

Elodie Bouzekri
ICS-IRIT, University Toulouse III
31062,Toulouse, France
Elodie.Bouzekri@irit.fr

ABSTRACT

Large companies rely on recommender systems to support users' processes of decision-making and analysis of large datasets of items. In critical context, such as civil aircraft cockpit, recommender systems can be a powerful tool for operators to support their tasks. Operators can be confronted with choosing the right option depending on the current alerts and context from a set of alternatives. The main goal of the presented PhD is to propose a model-based approach for the design and the development of dependable and usable recommender systems. This paper elaborates on challenges and approaches for engineering dependable and usable recommender systems.

https://doi.org/10.1145/3220134.3220147

KEYWORDS

Recommender Systems; Dependability; Usability; Modelling.

1 INTRODUCTION TO THE CONTEXT OF THE PHD

Recommender Systems (RS) are software tools and techniques that suggest items to a particular user (or a group of users), according to his preferences or needs [22]. Recommender systems are widely used in e-commerce or home entertainment (such as Netflix [7]) to enable the user to explore items of interest, consult details about the choices offered and help user to make a choice. These user tasks are relevant in other domain such as critical context where the operator has to deal with a lot of information to find potential solutions and take decision. Because, the cost of critical systems potential error is much higher than the costs of design and development [19], they have to be highly dependable and usable. We study how to adapt processes, methods and tools of critical software engineering to recommender systems engineering such as model-based approaches. Model-based approaches permit to describe in a complete and unambiguous way system behaviour, and then to make its behaviour predictable. In addition, we combine them with HCI engineering practices to improve the usability of these systems. Finally, the PhD thesis have the desire to extend these engineering practices to other kind of context with less need of dependability.

This doctoral consortium submission describes the PhD subject and its current development, describing a trail to dependable and usable recommender systems engineering. We propose to describe recommender systems and its behaviour using a formal model-based description dedicated to the description and design of interactive and dependable systems [18]. This work was previously aboard into two internships through prototyping critical recommender system and a proposition for a recommender system generic architecture.

2 RELATED WORK

2.1 Recommendation in Critical Context

In a critical context such as civil aircraft cockpit, in case of an alert, operators have to deal with being able to detect problems, identify causes, find potential solutions and take decision. However, there are only few contributions beyond the concept of a critical recommender system. [21] proposes a concept for a military recommender system to increase the effectiveness of a command body. Borg et al. [5] present a recommender system that proposes possible issue impacts in safety-critical context. However, these papers do not specify the architecture of the recommender system. In Decision Support Systems domain, that have a similar philosophy to recommender systems, we find several contributions in critical context such as [24] that present two decision support systems for the pilots of fighter aircraft or [23] that recommend the enemy fighter's future posture. Some work has also been done to develop decision support system for air traffic control like SKY-Scanner decision support system [14]. Nevertheless, like in recommender systems domains, these contributions do not present engineering aspects.

2.2 Engineering Recommender Systems

The main target for RS researcher has moved to other perceived quality of use such as user experience [13]. Nevertheless, engineering recommender systems is not yet a topic touched by academic research. As for the development of RS, several platforms have been proposed throughout the years like Lenskit [8] but requirements, specification, validation and verification and are not covered by generic programing platforms. Several mathematical formal models have been proposed to describe task of learning user preferences [12] or relationships [4]. However, to the best of our knowledge, there are no previous work on more advanced formal modeling (automata or petri net) of recommender systems and no work done on user task modeling and human error. There are no existing tool for this work.
These approaches are not sufficient to design and develop dependable and usable recommender systems because the engineering aspect of recommender systems is not enough developed. In the following, we present existing approaches for engineering interactive systems and dependable interactive systems.

3 EXISTING APPROACHES FOR THE DEVELOPMENT OF DEPENDABLE AND USABLE INTERACTIVE SYSTEMS

3.1 ° Modelling Interactive Systems

An important activity in software engineering is to modelling the system and the interactions. To model the system, we have to define its architecture. A well-defined architecture can reduce the coupling between each elements and improve maintainability for example. Defining an architecture is to specify all the architecture components, their behaviour, and their interactions

Table 1. Dependability attributes defined in [2].

Dependability Attribute	*Definition*
Availability	Readiness for correct service
Reliability	Continuity of service
Safety	Absence of catastrophic consequences on the user(s) and the environment
Integrity	Absence of improper system alterations
Maintainability	Ability to undergo modifications and repairs

between them, the environment and the user(s). In addition, the data manipulated have to be defined. Seeheim Model [20], Arch Model [3] or Pie and Red Pie [9] are examples of proposed interactive systems architectures. In addition, several user-centered engineering processes have been proposed to cover all the life cycle of these systems such like [11]. Recommender systems are interactive systems. In consequence, modelling only the system is incomplete. We have to model user tasks as well. Modelling user tasks brings many advantages like help to validate users' needs, help to design user interfaces and interactions, help to determine which tasks must or mustn't be automated and help to identify potential sources of human error (the exhaustive list can be found in [15]).

3.2 ° Development processes for dependable interactive systems

Building dependable systems requires identifying threats that can impair the functioning of the system to ensure his dependability. A system is dependable when it validates five attributes [2]: availability, reliability, safety, integrity and maintainability defined in Table 1. We have to adapt methods to ensure dependability [2] to recommender systems engineering embedded in a zero default approach (specification and verification). We will do not treat all issues but those that can be addressed by design. To deal with these issues we must build a process dedicated to avoid these issues for recommender systems. Process for systems with high-need of dependability existed like DO-178C supplement 330 [10] that provides an analysis process for avionics software systems. From these kind processes combine with human-centered process, we can propose a process like [15] for dependable and usable recommender systems.
However, these existing approaches do not take into account specificities of recommender systems such as the design of the context model component [1]. In addition, these approaches do not permit to resolve the predictability problems of the machine learning algorithms that the recommender systems embed.

4° AN APPROACH FOR THE DESIGN AND DEVELOPMENT OF DEPENDABLE AND USABLE RECOMMENDER SYSTEM

4.1 ° A Model-based Systematic Approach

To deal with this engineering problem, we propose a model-based systematic approach. This approach proposes to describe in a complete and unambiguous way the system, interactions between software component and between the user and the system. We propose to use a formal notation based on Petri nets [18] named ICO models. In order to describe the behaviour of each components of the recommender systems architecture. To avoid human error, to identify relevant

tasks to automate and to avoid automation surprises we should describe in a complete and unambiguous way user tasks. We propose to model user tasks with HAMSTERS [16], an approach similar to CTT [17] to show the synergy between recommender system behaviour and user interactions.

4.2 Adapt HCI methods to recommender systems engineering

To ensure the usability of recommender systems, we have to specify HCI behaviour of recommender system such as animations, visualizations, interaction techniques, etc. In addition, the complete and unambiguous description of the system and user tasks have the goal to define replicable recommender systems and to improve usability evaluation of these systems. Indeed, evaluate usability of a dynamic system (evolution of recommendation over time) is a difficult task. We think that more predictable recommender systems will help to the correct interpretation of these evaluations.

5 PROGRESS

Before the beginning of the PhD, I followed two internships with subjects related to the PhD subject. During the first one, I prototyped critical recommender system interfaces and during the second I started to work on recommender systems architecture. We have proposed the definition of a generic architecture for recommender systems (journal article accepted) [6]. We currently instantiate this architecture (see Figure 1) with two cases studies: an industrial study in critical context and a game.

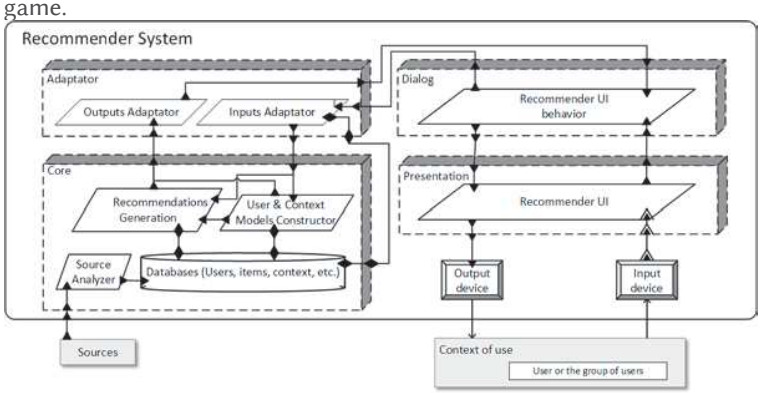

Figure 1: High-level View of a Proposition for Generic Architecture of Recommender Systems.

In addition, we start to define rules to apply for the design and the development of a critical recommender system. Another work in progress is to define formally automation in recommender systems thanks to HAMSTERS [16]. At the same time as this work, we search how to support better by design key usability criteria such as user control and explanation of recommendation.

REFERENCES

[1] Gediminas Adomavicius and Alexander Tuzhilin. 2015. Context-Aware Recommender Systems. In Recommender Systems Handbook. Springer, Boston, MA, 191–226.

[2] Avizienis, A., Laprie, J.-C., Randell, B., Landwehr, C. Basic concepts and taxonomy of dependable and secure computing. In IEEE Trans. on Dependable and Secure Computing, vol.1, no.1, pp. 11- 33, Jan.-March 2004.

[3] L. Bass, R. Little, R. Pellegrino, S. Reed, R. Seacord, and S. Sheppard. 1992. The Arch model: Seeheim revisited (version 1.0). In The UIMS Developers Workshop.

[4] Robert Bell, Yehuda Koren, and Chris Volinsky. 2007. Modeling Relationships at Multiple Scales to Improve Accuracy of Large Recommender Systems. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '07), 95–104..

[5] M. Borg, K. Wnuk, B. Regnell, and P. Runeson. 2016. Supporting Change Impact Analysis Using a Recommendation System: An Industrial Case Study in a Safety-Critical Context. IEEE Transactions on Software Engineering PP, 99: 1–1.

[6] E. Bouzekri, A. Canny, C. Fayollas, , C. Martinie, P. Palanque, E. Barboni, Y.Deleris, C. Gris. Engineering Issues Related to the Development of a Recommender System in a Critical Context: Application to Interactive Cockpits in International Journal on Human Computer Studies, ELSEVIER, accepted paper to appear in 2018.

[7] Carlos A. Gomez-Uribe and Neil Hunt. 2015. The Netflix Recommender System: Algorithms, Business Value, and Innovation. ACM Trans. Manage. Inf. Syst. 6, 4, Article 13 (December 2015), 19 pages

[8] Michael D. Ekstrand, Michael Ludwig, Joseph A. Konstan, and John T. Riedl. 2011. Rethinking the Recommender Research Ecosystem: Reproducibility, Openness, and LensKit. In Proceedings of the Fifth ACM Conference on Recommender Systems (RecSys '11), 133–140.

[9] Alan John Dix. 1991. Formal methods for interactive systems. Academic Press, London.

[10] DO-333, Formal methods supplement to do-178c and do-278a (December 2011).

[11] Bengt Göransson, Jan Gulliksen, and Inger Boivie. 2003. The usability design process – integrating user-centered systems design in the software development process. Software Process: Improvement and Practice 8, 2: 111–131.

[12] Sung Young Jung, Jeong-Hee Hong, and Taek-Soo Kim. 2002. A formal model for user preference. In 2002 IEEE International Conference on Data Mining, 2002. Proceedings., 235–242.

[13] Bart P. Knijnenburg, Martijn C. Willemsen, Zeno Gantner, Hakan Soncu, and Chris Newell. 2012. Explaining the User Experience of Recommender Systems. User Modeling and User-Adapted Interaction 22, 4–5: 441–504.

[14] Kristina Lapin, Vytautas Čyras, and Laura Saviciene. 2010. Visualization of Airport Procedures in Time Critical Decision Support Systems. 408–421.

[15] Célia Martinie, Philippe Palanque, David Navarre, and Eric Barboni. 2012. A development process for usable large scale interactive critical systems: application to satellite ground segments. In Proceedings of the 4th international conference on Human-Centered Software Engineering (HCSE'12), Marco Winckler, Peter Forbrig, and Regina Bernhaupt (Eds.). Springer-Verlag, Berlin, Heidelberg, 72-93.

[16] Martinie C., Palanque P., Winckler M. Structuring and Composition Mechanisms to Address Scalability Issues in Task Models. IFIP TC 13 INTERACT conference, (2011) 589-609, Springer Verlag.

[17] G. Mori, F. Paterno, and C. Santoro. 2002. CTTE: support for developing and analyzing task models for interactive system design. IEEE Transactions on Software Engineering 28, 8: 797–813.

[18] David Navarre, Philippe Palanque, Jean-Francois Ladry, and Eric Barboni. 2009. ICOs: A model-based user interface description technique dedicated to interactive systems addressing usability, reliability and scalability. ACM Trans. Comput.-Hum. Interact. 16, 4, Article 18 (November 2009), 56 pages.

[19] Philippe Palanque. & Rémi Bastide.1994. A Formalism for Reliable User Interfaces. Workshop Software Engineering / Human Computer Interaction associated with the IEEE / ICSE 16 conference. Sorento, Italy 16-21 May 1994.

[20] Günther. E. Pfaff (Ed.). 1983. User Interface Management Systems, Proc. Workshop on User Interface Management Systems, Springer Verlag.

[21] M. G. Pilarski. 2014. The Concept of Recommender System Supporting Command and Control System in Hierarchical Organization. In 2014 European Network Intelligence Conference, 138–141.

[22] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2015. Recommender Systems: Introduction and Challenges. In Recommender Systems Handbook. Springer, Boston, MA, 1–34.

[23] T. y Sun, S. j Tsai, Y. n Lee, S. m Yang, and S. h Ting. 2006. The Study on Intelligent Advanced Fighter Air Combat Decision Support System. In 2006 IEEE International Conference on Information Reuse Integration, 39–44.

[24] Peter Svenmarckt and Sidney Dekker. 2003. Decision support in fighter aircraft: from expert systems to cognitive modelling. Behaviour & Information Technology 22, 3: 175–184.