

Assessing Subjective Quality of Web Interaction with Neural Network as Context of Use Model

Maxim Bakaev^(✉), Vladimir Khvorostov, and Tatiana Laricheva

Novosibirsk State Technical University, Novosibirsk, Russia
{bakaev, xvorostov}@corp.nstu.ru, tatka_all1994@mail.ru

Abstract. Despite certain advances in automation of software quality assurance, testing and debugging remain the most laborious activities in the software development cycle. Evaluation of web interaction quality is still largely performed with traditional human effort-intensive methods, particularly due to the inevitable association of website usability with particular contexts of use, target users, tasks, etc. We believe that testing automation in this field may ultimately lead to better online experience for all and are important in promoting e-society development. We propose to employ artificial neural networks to predict website users' subjective impressions, whose importance is widely recognized but that are somehow overshadowed by the effectiveness and efficiency dimensions. We justify the structure of the network, with the input layer reflecting context of use, while the output layer consisting of the subjective evaluation scales (*Beautiful, Evident, Fun, Trustworthy, and Usable*). The experimental session with 82 users and 21 university websites was undertaken to collect the evaluation data for the network training. Finally, we verify the validity of the model by comparing it to a certain baseline, analyze the importance of the input factors, and provide recommendations for future evaluations-collecting sessions.

Keywords: User satisfaction · Methods for software analysis · Usability engineering · Testing automation

1 Introduction

As e-society develops, the swift advance of web-based software brought a new dimension to the classic software engineering process, with the extreme growth in the number of websites, the pressure for rapid development, the possibility of frequent updates, etc. At the dawn of the online era, most of web services and applications had relatively low complexity and were developed by small and/or inexperienced teams. This has clearly changed, as websites are becoming not only numerous and ubiquitous, but also complex, as basically any software is turning into web software. The special feature of web interfaces that shape the modern e-society is that their users are by and large impatient novices: web users are experienced with neither website in particular, and they leave easily, unless their very first impressions of the website are positive and pleasant ones. Let us then recount which popular web software quality assurance (QA) techniques currently address this issue and if their automation is done.

Testing visual appearance of a page and ensuring its reasonable correspondence to the design prototype is yet another type of testing, specific for the web. In principle, screenshots of the page rendered in various browsers can be auto-compared to design mockups, but in practice this is rarely done, due to complexity and ambiguity of image analysis tasks. In contrast, **load testing** is naturally automatable, since high number of relatively unsophisticated requests can be generated by a dedicated tool straightforwardly and quickly, unlike by manual workforce. As for **usability and interaction quality testing** automation, it does use approaches and tools of the functional-, visual appearance-, and load testing – but to a limited coverage. Indeed, a usable application **necessarily** corresponds to the functional and design specifications and responds promptly, but this isn't **sufficient** for its high usability. Meanwhile, the advance of web-based software is due not just to its universal availability, maintainability, and other similar factors, but is also associated with the technological development and standardization of web interfaces.

In contrast to most other software quality attributes, such as reliability, maintainability, etc. [1], interaction quality is **relative** to a context of use (see e.g. the ISO 9241-11 1998 specification for usability), which obviously thwarts attempts towards its testing automation. It is not by chance that existing techniques are mostly able to assess user- and task-independent quality aspects, such as e.g. validating correctness of HTML code or its correspondence to accessibility guidelines [2], but not attributes like learnability (being easy to grasp and start using a product), attractiveness, satisfaction, etc. Although there remains certain conceptual ambiguity both in literature and in practical QA in respect to usability (see our review in [3]), its three commonly recognized dimensions are identified as **effectiveness**, **efficiency**, and **satisfaction**. The subjective dimension of satisfaction is still somehow disfavored compared to the other two, even though the role of aesthetic impression, trust, pleasure, etc. is widely recognized for websites' success. Also, for better or for worse, qualitative usability evaluations prevail, understandably hindering automation and leading to unavoidable involvement of costly usability engineers, and the results depend heavily of their expertise level. Meanwhile, employment of real users, experts or specialists is not always the most effective way, especially if evaluations are needed for a great number of website versions. For example, in web engineering based on evolutionary algorithms, repeated assessment of the candidate solutions' quality would be unfeasible through interactive means only (i.e. by humans), and introduction of certain computable fitness function representing the environment (i.e. context of use) is essential [4].

So, in our paper we consider the problem of automated assessment of web interaction subjective quality – i.e. predicting user's subjective evaluation of a website **without an actual user**. Among the advantages of automated evaluation of website usability the following ones are commonly noted [5]: lower costs, reduction of human expertise needed, better consistency and coverage, emerging capabilities to predict losses from usability problems and to promptly evaluate different design versions, etc. Thus we believe that advances in this field may ultimately lead to better online experience for all and are important in promoting e-society development. Section 2 is dedicated to general overview of web usability evaluation and assessment methods, with special focus on artificial neural networks (NN), which is the method we apply in

our current work. Particularly, we explain about potential feasibility of the method and justify the structure of the network, with the input layer reflecting context of use, while the output layer consisting of popular subjective evaluation scales. In Sect. 3, we describe how we collected the evaluations and then constructed and trained the network. We also verify the validity of the model by comparing it to a baseline and analyze the importance of the input factors, providing recommendations for scheming future evaluation-collecting sessions.

2 Methods

Before anything else, we'd like to note that since there seems to be little consensus in conceptual application of “quality evaluation” and “quality assessment” terms, we are going to use them as close synonyms, the former rather relating to approximate values and only certain aspects of quality, while the latter being more quantitative and involving more rigorous process. Also, we equal “interaction quality” to “usability” in the assumption that functional requirements are met and the **effectiveness** dimension of the latter is ensured. In the current section, we provide overview of existing methods and tools for web usability assessment and engineering, and then describe the proposed neural network-based approach.

2.1 Traditional and Automated Web Interaction Quality Assessment

Though it is said that “Each method [for assessing usability]... is unique, and relies heavily on the skills and knowledge of evaluators” [6], the set of web usability evaluation methods universally includes the following kit [3]:

- **User observation** – may be explicit, i.e. watching real users performing real tasks in a real context, or implicit, such as analyzing logs or even videos of user behavior on a website. Major web analytics services (Yandex.Metrica, Google Analytics, etc.) increasingly provide the means for tracking and analyzing user interactions with respect to usability.
- **Usability testing** – while both users and tasks are “replicated” to match the real ones, the method is consistently listed among the most effective methods in usability engineering. Lately, as broadband channels became capable of streaming video in real time, numerous web services for performing remote usability testing emerged (Userlytics, OpenHallway, etc.), many of which also aid in recruiting participants and generally allow saving costs and time.
- **Surveys and inspections** – either with real users (e.g. with a feedback form) or with usability experts (heuristic evaluation), but without guaranteed fulfillment of the actual tasks. Nowadays, free or freemium services such as Usabilityhub, Usabilla, or Askusers.ru aid in specifying usability-related questions and obtaining the answers from users. Heuristic evaluation can be semi-automated via specifying checklists of design guidelines/heuristics and even auto-validating those that are user- and task-independent.

The interest towards automating web interaction quality assessment increased in the 2000s [7], together with dramatic growth of the online economy and the number of websites. Currently, the respective approaches may be divided into the following major groups (see more detailed review in [3]), summarized as shown in Table 1:

Table 1. Approaches towards automated web interaction quality assessment

Approach	Users, website, interaction	Limitations
Interaction-based	Real or representative users Real or prototype website (dynamic) Real interaction (reflected in logs)	In scope – mostly “owned” websites In pace – enough users must visit In disservice – potential losses from poor quality
Metric-based	No users Real website (static) No interaction (code or design analysis)	In generalization – metrics and their weights are specific to use context In expertise – may need human assessors/annotators
Model-based	Model or simulated users Model, prototype, or real website Simulated interaction, analysis of other interactions’ data	In resources – collected data, up-to-date models, computational powers In knowledge – models must be created and trained

- **Interaction-based** ones use data obtained from real or test interactions with the assessed website. These may involve analysis of mouse actions, keyboard inputs, “optimality” of user paths on website, detection of “usability threats” [8], etc.
- **Metric-based** – rely on operational website characteristics that presumably reflect its usability. These are generally extracted from available website source code or design [9], with subsequent quantitative evaluations or comparisons against “good practices” – established design guidelines (see e.g. [2] or [7]).
- **Model-based** – seek to obtain usability evaluations from models (mainly Domain, User, and Tasks [10]) as well as through general knowledge about human behavior and web technologies [11].

Thus, no single method or tool is the ultimate solution (lack of the universally accepted operationalization of usability surely adds to this as well), so combining them often yields better results, particularly in terms of ROI. The hybrid approaches often rely on AI and machine learning methods, which may include real-time processing of interaction data (including eye-tracking or brainwave analysis), model-based evolutionary approaches [12], usability models trained with data obtained from both interaction and surveys [13], and so on. The possible outcome ranges from merely the assessed quality value to identification of potential usability problems [8], to linking to relevant design guidelines [14], to even on-the-fly augmentation (re-generation) of the interface [15]. So, in our current work we are considering the application of artificial neural networks, which are generally recognized as a promising method in software testing [16], to assess subjective quality of web interaction.

2.2 Neural Network-Based Web Interaction Quality Assessment

Basically, a NN is a sophisticated way of specifying a function, and they are naturally used in increasingly popular evolutionary algorithms to specify fitness function that evaluates the quality of the candidate solutions, at least preliminarily [17]. NNs also have very reasonable computational effectiveness compared to other AI or statistical methods and are self-adapting, which allows accommodating the problem of ever-changing software requirements [18]. NNs are first trained and then tested on real data, attempting to generalize the obtained knowledge in classification, prediction, decision-making, etc. The available dataset is generally partitioned into training, testing, and holdout samples, where the latter is used to assess the constructed network – estimate the predictive ability of the model. The network performance is estimated via percentage of incorrect predictions (for categorical outputs) or relative error that is calculated as sum-of-squares relative to the mean model (the “null” hypothesis).

Neural networks have long history in software testing automation, but they generally focus on functional requirements. A notable “social” exception is the works applying the popular Kansei Engineering method in web design, which establish connections between design factors (input neurons) and users’ subjective evaluations per emotional scales (output neurons) [9, 19]. However, this “emotional engineering” has no emphasis on user interaction, so for our purposes, i.e. prediction of web interaction subjective quality, it makes more sense to have quality attributes as output neurons. The input, since usability is an “emergent property of usage” [6], should necessarily include factors of use context – i.e. User, Platform, and Environment attributes. The final consideration is the diversity of input data that is essential for proper NN training [16] and thus its ultimate feasibility for the quality prediction, so the input factors’ importance should be studied to better design future data collection sessions.

In our current work, we are going to use websites from a single domain and a relatively uniform target user group. So, the proposed structure of the NN input is the following:

- **User-related** factors: age, gender, and language/cultural group. The selection of the factors is based on general knowledge of important user attributes. We did not include the user experience factor, since (a) it’s hard to identify unambiguously, and (b) it would rather define a separate target user group in our settings.
- **Platform-related** factors: website country group, number of website sections (major chapters), Flesch-Kincaid Grade Level (as assessed by <https://readability-score.com>), and number of errors plus warnings (reported for a website by <https://validator.w3.org> code validator).
- **Environment-related** factors: page load time, global rank in terms of visitors, and bounce rate (all of these provided by <http://alexa.com>).

The NN output is the following evaluation scales representing popular dimensions of subjective interaction quality: *Beautiful*, *Evident*, *Fun*, *Trustworthy*, and *Usable*. In the following section we describe the experimental research that we undertook to collect data, create and train the NN, and estimate the accuracy of the model.

3 Experimental Results

3.1 The Experimental Setup

For the assessment, we employed 21 websites, all relating to a single fixed domain – *Career and Education*: 11 websites of German universities and 10 of Russian ones (for both groups, English versions were used), with sufficiently diverse designs with respect to layout, colors, images, etc. The website country factor was assigned two possible levels: German and Russian. The evaluators were 82 students from a German and a Russian university, virtually all of them majoring in Computer Science/Informatics, but the level of the program, i.e. Bachelor or Master, was not included as an input factor. More detailed data on the participants are presented in Table 2.

Table 2. The data on the two user groups in the experiment

Total number of participants		German users	Russian users
		40	42
Gender	Male	90%	71.4%
	Female	10%	28.6%
Program	Bachelor	35%	54.8%
	Master	65%	45.2%
Age	Range	19–33	20–28
	Mean	24.5	21.7
	SD	3.19	0.89
Native language	Common	German: 75%	Russian: 90.5%
	Others	25%	9.5%

The language group of the participants was assigned 3 levels: German, Russian and Other (Chinese, Arabic, Turkish, Yakut, Mongolian, or Kazakh). The experiment was performed at the universities’ computer rooms in several sessions, with participants using diverse equipment to evaluate the websites through specially developed web software (<http://ks.khvorostov.ru>). Each user was asked to evaluate 10 websites randomly selected from the 21 and presented in random order, by the five subjective scales, with values for each ranging from 1 (worst) to 7 (best).

3.2 Descriptive Statistics

In total, data sets for 820 full evaluations of websites were recorded in the experimental software database. The mean evaluations with standard deviations per the scales are presented in Table 3 (SPSS software was used for statistical analysis and construction of the NN). We applied ANOVA to assess statistical significance of differences between evaluations by the two user groups – the significant ones at $\alpha = .06$ are marked in bold, while the p-values are provided in the respective column. In Table 4 we present pair-wise Pearson correlations between the user evaluations per the subjective quality scales – all correlations are significant ($p < 0.001$).

Table 3. Mean (SD) evaluations per the scales for the two user groups

Scale	German users	Russian users	Difference
Beautiful	3.83 (0.59)	4.10 (0.98)	
Trustworthy	4.10 (0.37)	4.53 (0.76)	p = 0.058
Fun	3.54 (0.50)	4.20 (0.97)	p = 0.025
Evident	3.79 (0.27)	4.60 (0.66)	p < 0.001
Usable	3.78 (0.36)	4.52 (0.79)	p = 0.003

Table 4. Correlations between the subjective quality scales

Scales	Beautiful	Trustworthy	Fun	Evident	Usable
Beautiful	1	0.670	0.767	0.563	0.605
Trustworthy		1	0.673	0.648	0.675
Fun			1	0.638	0.661
Evident				1	0.813
Usable					1

More complete description of the data can be found in the relevant technical report [20].

3.3 The Neural Network

The neural network was constructed in SPSS with the following settings: *Multilayer Perceptron*, 1 hidden layer with 10 neurons (half of the sum neurons of input layer and output layer), optimization algorithm – *gradient descent*, activation functions – *sigmoid* for the hidden layer and *identity* – for output layer. The partition of the data was 80% for training data, 10% – test, 10% – holdout (81 evaluations). We transformed the output scales from Ordinal to Scale, since we were interested in average predicted evaluations rather than their exact values. The average overall relative errors for the resulting NN model are presented in Table 5.

Table 5. Average overall relative errors for the subjective quality scales

Scale	Error
Beautiful	0.656
Fun	0.707
Evident	0.762
Trustworthy	0.786
Usable	0.792
Model	0.737

The average overall relative error value of 0.737 does not allow us to conclude that the constructed NN model is by all means good in predicting subjective evaluations of

users. The errors are quite diverse for different scales, and seemingly the model better predicts impressions for the scales that users understand clearer: compare 0.656 for *Beautiful* and 0.707 for *Fun*, quite common terms, versus 0.786 for *Trustworthy* and 0.792 for *Usable*, which are much more specialized. This highlights again the necessity to pay close attention to exact names of the subjective scales (the fact known well in Kansei Engineering) and calls for the use of simpler and less ambiguous terms.

As a baseline for the NN-induced assessment accuracy, we'd like to use the data and the algorithm from one of our previous works [21], in which we employed a guideline-based method for website quality evaluation, with 24 users and 6 e-commerce websites. The subjective quality values assessed with the proposed fuzzy relations algorithm based on correspondence to guidelines and the actual user evaluations obtained via usability testing sessions are presented for comparison in Table 6.

Table 6. Assessed and evaluated subjective interaction quality for the websites

Website id	Website description	Assessed quality	User evaluation
#1	A web forum for pensioners	64.97%	62.99%
#2	A non-state pension fund	77.01%	66.07%
#3	A business education center	65.28%	82.32%
#4	An online medical shop	62.35%	72.95%
#5	A web development company	78.26%	74.62%
#6	An educational website for seniors	88.03%	87.50%
Average:		72.65%	74.41%

The correlation between the quality values assessed by the algorithm and evaluated by users is thus relatively low, at 0.448, but we would also like to obtain the relative errors for the comparison. In calculating the error, as the null model we propose to use the average value of assessed quality, since users are known to be often biased in their evaluations and the “real null” of 50%, i.e. the truly average value between the worst and the best value in the scale, is rarely reported in studies. The relative error value calculated this way equals to 1.066, which is considerably worse than the 0.737 we got for the NN model. We'd like to note, however, that this example is only given as an illustrative baseline, but not to actually assess quality of any websites, since the number of employed users, guidelines and websites was too low, and the training set in that research was not different from the assessed set of websites. In Table 7 we present the results of the importance analysis for the independent variables (inputs).

We can see that the factors related to *User*, *Platform*, and *Environment* dimensions of the use context are quite mixed in terms of their importance. Still, some potentially useful recommendations could be made for future evaluation sessions (within the considered domain of *Career and Education*):

- When selecting users, make sure that age diversity is maintained in the sample, but it's hardly necessary to represent both genders, since the latter factor was not important with respect to the evaluations provided.

Table 7. The independent variables' (factors') importance

Factor	Importance	Normalized importance
Flesch-Kincaid grade level	0.177	100.0%
User age	0.161	90.9%
Alexa bounce rate	0.135	75.9%
Page load time	0.130	73.0%
Website group	0.109	61.4%
Errors + Warnings	0.105	59.4%
User language group	0.069	39.0%
Alexa global rank	0.054	30.6%
Number of sections	0.044	24.6%
User gender	0.016	8.8%

- For cross-cultural generalization to be feasible, it is important to employ websites from different countries and users of different cultures.
- The employed websites don't have to be different in scale (reflected as the number of sections and number of visitors in our experiment), but the diversity of content (grade level) and of technical quality (response speed, code validity) should be maintained.

4 Conclusions

Web interfaces continue to gain popularity in software engineering and their development and testing constitute a very significant share in modern web projects – in terms of not just website design, but the whole web interaction quality assurance. Analysis and testing automation, which is widely recognized as very much desirable, in this field implies assessing usability without users or experts, since though websites are abundant, human effort and time are not. In our paper, we explored capabilities of an artificial neural network in predicting subjective evaluations of a website, within a fixed domain (university websites) and target user group (students). For that, we conducted an experimental session with 82 users and 21 websites, collecting 820 full evaluations and using them to train the NN (80%) and estimate the predictive ability of the model.

The relative errors (average 0.737) suggest moderate predictive potential of the model, which however was better than the fuzzy relations algorithm from one of our previous works used as a baseline for comparison [21]. The “common-sense” subjective evaluation scales (*Beautiful* and *Fun*) had considerably lower errors compared to more complex and specialized ones (*Usable* and *Trustworthy*). So, one may be recommended to ensure that the employed adjectives are clear and unambiguous for the users. The analysis we performed with the context of use factors (*User*, *Platform*, and *Environment* dimensions) showed their mixed importance, and we provided recommendations on user and website selection in future evaluation data gathering sessions.

Among the limitations of our research are fixed domain of websites and a single target user group. It also remained unclear whether the obtained number of evaluations

was reasonable, or should more data be collected for the network training. The prospects of our future research work should include varying the amount of richer training data with respect to users and websites and analyzing the quality of the resulting NN models.

Acknowledgement. The reported study was funded by RFBR according to the research project No. 16-37-60060 mol_a_dk. The authors also thank M. Gaedke and S. Heil from TU Chemnitz who facilitated the collection of user evaluations.

References

1. Glass, R.L.: *Facts and Fallacies of Software Engineering*. Addison-Wesley Professional, Boston (2002)
2. Gay, G., Li, C.Q.: AChecker: open, interactive, customizable, web accessibility checking. In: 2010 International Cross Disciplinary Conference on Web Accessibility (W4A 2010), no. 23 (2010)
3. Bakaev, M., Mamysheva, T., Gaedke, M.: Current Trends in automating usability evaluation of websites. In: International Forum on Strategic Technologies (IFOST), NSTU, pp. 510–514 (2016)
4. Bakaev, M., Gaedke, M.: Application of evolutionary algorithms in interaction design: from requirements and ontology to optimized web interface. In: IEEE EIConRusNW, pp. 125–130. LETI, St. Petersburg (2016). doi:[10.1109/EIConRusNW.2016.7448138](https://doi.org/10.1109/EIConRusNW.2016.7448138)
5. Dingli, A., Mifsud, J.: Useful: a framework to mainstream web site usability through automated evaluation. *Int. J. Hum. Comput. Interact. (IJHCI)*, **2**(1), 10–30 (2011)
6. The Encyclopedia of Human-Computer Interaction, 2nd edn. (2016). <https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed>. Accessed 07 Apr 2017
7. Vanderdonckt, J., Beirekdar, A., Noirhomme-Fraiture, M.: Automated evaluation of web usability and accessibility by guideline review. In: Koch, N., Fraternali, P., Wirsing, M. (eds.) ICWE 2004. LNCS, vol. 3140, pp. 17–30. Springer, Heidelberg (2004). doi:[10.1007/978-3-540-27834-4_4](https://doi.org/10.1007/978-3-540-27834-4_4)
8. Grigera, J., Garrido, A., Rivero, J.M., Rossi, G.: Automatic detection of usability smells in web applications. *Int. J. Hum.-Comput. Stud.* **97**, 129–148 (2017)
9. Lin, Y.C., Yeh, C.H., Wei, C.C.: How will the use of graphics affect visual aesthetics? A user-centered approach for web page design. *Int. J. Hum.-Comput. Stud.* **71**(3), 217–227 (2013)
10. Feuerstack, S., Blumendorf, M., Kern, M., Kruppa, M., Quade, M., Runge, M., Albayrak, S.: Automated usability evaluation during model-based interactive system development. In: Forbrig, P., Paternò, F. (eds.) HCSE/TAMODIA-2008. LNCS, vol. 5247, pp. 134–141. Springer, Heidelberg (2008). doi:[10.1007/978-3-540-85992-5_12](https://doi.org/10.1007/978-3-540-85992-5_12)
11. Palanque, P., Barboni, E., Martinie, C., Navarre, D., Winckler, M.: A model-based approach for supporting engineering usability evaluation of interaction techniques. In: 3rd ACM SIGCHI symposium on Engineering Interactive Computer Systems, pp. 21–30 (2011)
12. Pleuss, A., Wollny, S., Botterweck, G.: Model-driven development and evolution of customized user interfaces. In: 5th ACM SIGCHI Symposium on Engineering Interactive Computing Systems, pp. 13–22 (2013). doi:[10.1145/2494603.2480298](https://doi.org/10.1145/2494603.2480298)

13. Speicher, M., Both, A., Gaedke, M.: Ensuring web interface quality through usability-based split testing. In: Casteleyn, S., Rossi, G., Winckler, M. (eds.) ICWE 2014. LNCS, vol. 8541, pp. 93–110. Springer, Cham (2014). doi:[10.1007/978-3-319-08245-5_6](https://doi.org/10.1007/978-3-319-08245-5_6)
14. Viswanathan, S., Peters, J.C.: Automating UI guidelines verification by leveraging pattern based UI and model based development. In: ACM CHI 2010 EA on Human Factors in Computing Systems, pp. 4733–4742 (2010). doi:[10.1145/1753846.1754222](https://doi.org/10.1145/1753846.1754222)
15. Barrera-León, L., et al.: TUKUCHIY: a dynamic user interface generator to improve usability. *Int. J. Web Inf. Syst.* **12**(2), 150–176 (2016)
16. Vanmali, M., Last, M., Kandel, A.: Using a neural network in the software testing process. *Int. J. Intell. Syst.* **17**(1), 45–62 (2002). doi:[10.1002/int.1002](https://doi.org/10.1002/int.1002)
17. Dias, J., et al.: A genetic algorithm with neural network fitness function evaluation for IMRT beam angle optimization. *Central Eur. J. Oper. Res.* **22**(3), 431–455 (2014)
18. Yao, X.: Evolving artificial neural networks. *Proc. IEEE* **87**(9), 1423–1447 (1999)
19. Guo, F., et al.: Optimization design of a webpage based on Kansei engineering. *Hum. Factors Ergon. Manuf. Serv. Ind.* **26**(1), 110–126 (2016). doi:[10.1002/hfm.20617](https://doi.org/10.1002/hfm.20617)
20. Bakaev, M., Gaedke, M., Heil, S.: Kansei engineering experimental research with university websites. Technical report, TU Chemnitz, CSR-16-01 (2016)
21. Bakaev, M., Avdeenko, T.: User interface design guidelines arrangement in a recommender system with frame ontology. In: Yu, H., Yu, G., Hsu, W., Moon, Y.-S., Unland, R., Yoo, J. (eds.) DASFAA 2012. LNCS, vol. 7240, pp. 311–322. Springer, Heidelberg (2012). doi:[10.1007/978-3-642-29023-7_31](https://doi.org/10.1007/978-3-642-29023-7_31)