Ensuring Web Interface Quality through Usability-based Split Testing

Maximilian Speicher[1,2], Andreas Both[2], and Martin Gaedke[1]

[1]Chemnitz University of Technology, 09111 Chemnitz, Germany
{maximilian.speicher@s2013,martin.gaedke@informatik}.tu-chemnitz.de
[2]R&D, Unister GmbH, 04109 Leipzig, Germany
{maximilian.speicher,andreas.both}@unister.de

Abstract. Usability is a crucial quality aspect of web applications, as it guarantees customer satisfaction and loyalty. Yet, effective approaches to usability evaluation are only applied at very slow iteration cycles in today's industry. In contrast, conversion-based split testing seems more attractive to e-commerce companies due to its more efficient and easy-to-deploy nature. We introduce *Usability-based Split Testing* as an alternative to the above approaches for ensuring web interface quality, along with a corresponding tool called *WaPPU*. By design, our novel method yields better effectiveness than using conversions at higher efficiency than traditional evaluation methods. To achieve this, we build upon the concept of split testing but leverage user interactions for deriving quantitative metrics of usability. From these interactions, we can also learn models for predicting usability in the absence of explicit user feedback. We have applied our approach in a split test of a real-world search engine interface. Results show that we are able to effectively detect even subtle differences in usability. Moreover, WaPPU can learn usability models of reasonable prediction quality, from which we also derived interaction-based heuristics that can be instantly applied to search engine results pages.

Keywords: Usability, Metrics, Heuristics, Interaction Tracking, Search Engines, Interfaces, Context-Awareness

1 Introduction

In e-commerce, the usability of a web interface is a crucial factor for ensuring customer satisfaction and loyalty [18]. In fact, Sauro [18] states that "[p]erceptions of usability explain around 1/3 of the changes in customer loyalty." Yet, when it comes to interface evaluation, there is too much emphasis on so-called *conversions* in today's industry. A conversion is, e.g., a submitted registration form or a completed checkout process. While such metrics can be tracked very precisely, they lack information about the actual usability of the involved interface. For example, a checkout process might be completed accidentally due to wrongly labeled buttons. Nielsen [17] even states that a greater number of conversions can be *contradictory* to good usability. In the following, we illustrate this challenge by introducing a typical example scenario.

Scenario. A large e-commerce company runs several successful travel search engines. For continuous optimization, about 10 split tests are carried out per live website and week. That is, slightly different versions of the same interface are deployed online. Then, the one gaining the most conversions is chosen after a predefined test period. The main stakeholder, who studied business administration and founded the company, prefers the usage of *Google Analytics*[1] or similar tools due to their precise and easy-to-understand metrics. Yet, the split testing division would like to gain deeper insights into users' behavior since they know that conversions do not represent usability. Thus, they regularly request more elaborate usability evaluations, such as expert inspections for assessing the interfaces. The stakeholder, however, approves these only for novel websites or major redesigns of an existing one. To him, such methods—although he knows they are highly effective[2]—appear to be overly costly and time-consuming. *Conversion-based split testing* seems more attractive from the company's point of view and is the prime method applied for optimization.

Requirements. The situation just described is a common shortcoming in today's e-commerce industry, which is working at increasingly fast iteration cycles. This leads to many interfaces having a suboptimal usability and potentially deterring novel customers. Thus, we formulate three requirements for a novel usability testing approach that would be feasible for everyday use in industry and support a short time-to-market:

(R1) Effectiveness A novel approach must be more effective than conversion-based split testing w.r.t. determining the usability of an interface.

(R2) Efficiency A novel approach must ensure that evaluations are carried out with minimal effort for both, developers and users. Particularly, deployment and integration must be easier compared to established methods such as expert inspections or controlled lab studies.

(R3) Precision A novel approach must deliver precise yet easy-to-understand metrics to be able to compete with conversion-based split testing. That is, it must be possible to make statements like "Interface A has a usability of 99% and Interface B has a usability of 42%. Thus, 'A' should be preferred."

A solution to the above is to derive usability directly from interactions of real users, such as proposed by [21]. However, they conclude that user intention and even small deviations in the low-level structures of similar webpages influence interactions considerably. This makes it difficult to train an adequate *usability model* M that predicts usability U from interactions $\boldsymbol{I}$ only: $M(\boldsymbol{I}) = U$. Still, the described approach yields great potential.

We consider the pragmatic definition of usability as presented in [20], which is based on ISO 9241-11. Using a corresponding instrument specifically designed for correlation with client-side interactions [20], we propose a general approach to *Usability-based Split Testing* rather than considering conversions. To achieve

[1] `http://www.google.com/analytics/` (2014-02-01).

[2] For example, [16] state that only five evaluators can find up to 90% of the usability problems in an interface.

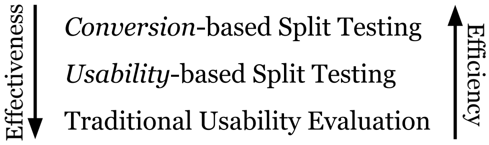

Effectiveness
Conversion-based Split Testing
Usability-based Split Testing
Traditional Usability Evaluation
Efficiency

Fig. 1. Web Interface Usability Evaluation: the competing approaches (rough overview).

this, we provide *WaPPU*—a tool that caters for (a) user interaction tracking, (b) collecting usability judgments from real users, (c) training usability models and (d) correlation of the obtained data. By design, the concept of Usability-based Split Testing enables developers to *ensure the quality of a web application* w.r.t. its interface usability at higher effectiveness than conversion-based split testing and higher efficiency than traditional approaches to usability evaluation (Fig. 1).

Making use of WaPPU ("*Wa*s that *P*age *P*leasant to *U*se?") we performed a usability-based split test of a real-world search engine results page (SERP). We paid specific attention to user intention and differences in low-level page structure to overcome the problems pointed out in [21]. From the study results, we derived interaction-based usability models and quantitative heuristic rules for SERPs. These can be instantly applied to user interactions collected on a SERP for a reasonable approximation of usability at very high efficiency.

In the following section, we give an overview of related work and describe the initial user study motivating our novel approach. Subsequently, we explain the concept of Usability-based Split Testing (Sec. 3) and the corresponding tool WaPPU (Sec. 4). The evaluation involving two web interfaces of a real-world search engine are presented in Section 5, followed by our findings (Sec. 6). Current limitations of our approach and potential future work are discussed in Section 7 before giving concluding remarks in Section 8.

2 Related Work

Our research is related to a wide variety of existing work. In particular, we are going to refer to *automatic* and *metrics-based* approaches to usability evaluation that are partly based on *user interaction analysis*. We also present an earlier study on the feasibility of quantitative interaction-based usability evaluation.

2.1 Automatic Approaches to Usability Evaluation

User Interaction Analysis Atterer et al. [1] present a tool for client-side user interaction tracking. After having collected information about cursor behavior, keyboard strokes or dwell time, one can use these events to visualize a users interactions on a webpage. From these, the authors aim to infer implicit interactions, such as hesitation before filling in an input field [1]. This is a useful tool for facilitating more automatic usability tests and provides developers with valuable information. *m-pathy*[3] is a commercial tool for qualitative user behavior inspection that follows the concept described by [1]. The tool features additional metrics that are, however, in analogy to conversion-based split testing, e.g., the number of checkout processes and similar.

Web Usability Probe [2] is a more sophisticated tool also allowing for automatic remote usability testing. It is possible to define optimal logs for given tasks, which are then compared to the client-side logs actually produced by the user. De Vasconcelos and Baldochi Jr. [4] follow a similar approach that compares users' interactions against pre-defined patterns.

In contrast to our novel approach, all of the above methods—although as well aiming at usability improvement—have different focuses. None derives *quantitative* statements about usability from the observed interactions, which would enable direct comparison of interfaces. Rather, interpretation of the delivered qualitative information is largely up to a developer or dedicated usability evaluator.

Navalpakkam and Churchill [12] investigate the possibility to infer the user experience of a web interface from mouse movements. In a user study, they find that certain features of interaction (e.g., hovers, arrival time at an element) can be used to predict reading experience and user distraction with reasonable accuracy. Yet, they investigate only these specific aspects. Particularly, the authors do not focus on providing interaction-based measures of usability or user experience for quantitative comparison of interfaces.

Website Checking Tools such as *AChecker* [6] and *NAUTICUS* [3] aim at automatic checking of websites according to certain criteria and heuristics. While the first specifically focuses on web accessibility, the second tool also takes into account usability improvements for visually impaired users. Both tools are particularly able to automatically suggest improvements regarding the investigated interfaces. Yet, they only consider static criteria concerned with structure and content of a website rather than actual users' interactions.

A/B Testing *AttrakDiff*[4] is a tool that enables A/B testing of e-commerce products for user experience optimization. That is, based on a dedicated instrument, the hedonic as well as pragmatic quality of the products are compared [11]. While this may seem very similar to our proposed approach, it has to be noted that the aim of AttrakDiff is different from Usability-based Split Testing. Particularly, the tool leverages questionnaire-based remote asynchronous evaluation rather than focusing on user interactions. Also, qualitative, two-dimensional statements about user experience are derived, which has to be clearly distinguished from usability and quantitative metrics thereof.

[3] `http://www.m-pathy.com/cms/` (2014-02-24).

[4] `http://attrakdiff.de/` (2014-02-24).

2.2 Metrics-based Approaches to Usability Evaluation

Contrary to the above approaches, Nebeling et al. [14] take a step into the direction of providing quantitative metrics for webpage evaluation. Their tool analyzes a set of spatial and layout aspects, such as *small text ratio* or *media–content ratio*. These metrics are static (i.e., purely based on the structure of the HTML document) and specifically aimed at large-screen contexts. In contrast, our goal is to provide *usability-in-use* metrics based on users' dynamic interactions with the webpage.

W3Touch [15] is a metrics-based approach to adaptation of webpages for touch devices. This means certain metrics of a website, e.g., *average zoom*, are determined from user interactions on a per-component basis. Components with values above a certain threshold are then assumed to be "critical" and adapted according to rules defined by the developer. This is a very promising approach that is, however, specifically aimed at touch devices. Moreover, the webpage metrics that identify potentially critical parts of a webpage are not transferred into more precise statements about usability.

2.3 Motivating Study

In the following, we address earlier work of the authors of this paper [21] that motivates the concept of Usability-based Split Testing.

In [21], we have tried to solve the already described conflict between traditional usability evaluations and conversion-based split testing by learning usability models from user interactions. For this, we have collected user interaction data on four similarly structured online news articles. Study participants had to answer a specific question about their assigned article. However, only two of the articles contained an appropriate answer. Once they found the answer or were absolutely sure the article did not contain it, participants had to indicate they finished their task and rate the web interface of the article based on the novel INUIT usability instrument [20]. INUIT contains seven items designed for meaningful correlation with client-side interactions (e.g., "cursor speed positively correlates with confusion") from which an overall usability score can be derived.

All articles featured a single text column with a larger image on top and a sidebar which contained additional information and/or hyperlinks. Two articles featured a short text ($\sim$1 page) while the remaining two featured a longer text ($\geq$2 pages). Moreover, two of the articles featured images within or close to the text column. Still, the high-level structures of the articles were similar. The lower-level differences were chosen by purpose to provoke differences in user behavior and usability judgments. Also, by providing an answer to the participant's task in only two of the four articles, we have simulated different user intentions according to [8]. That is, participants who could find an answer acted like *fact finders* while the remaining participants behaved like *information gatherers*.

We used a dedicated jQuery plug-in to track a set of well-defined interactions (e.g., clicks, hovers, scrolling distance etc.). These interactions were recorded separately for: (1) the whole page; (2) a manually annotated area of interest, i.e., the article text; (3) all text elements; (4) all media elements; (5) text elements within the area of interest; and (6) media elements within the area of interest.

To give a representative example, the interaction feature $hoverTime_{text}$ describes the aggregated time the user spent hovering text elements anywhere on the page. Furthermore, all interaction feature values were normalized using appropriate page-specific measures. For example, the page dwell time was divided by the length of the article text to ensure comparability across the four webpages.

The main hypothesis investigated in the study was *whether it is possible to learn a common usability model from similarly structured webpages.* Such a model should be able to also predict the usability of a webpage which did not contribute training data, as long as its structure is similar to a certain degree. However, when we correlated the collected interactions and usability judgments, we found huge differences between the four news articles. In fact, there was no interaction feature that showed a considerable correlation with usability for all four investigated webpages.

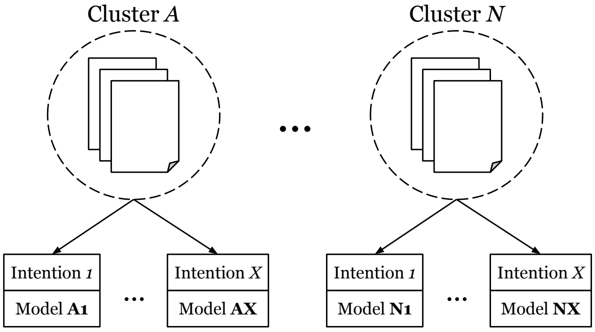


Fig. 2. Concept of a general framework for providing interaction-based usability models.

This result indicates that user behavior depends on low-level webpage structure and intention more strongly than assumed, i.e., *interactions* are a function of *usability*, *structure* and *intention*.

Thus, we concluded that a general framework for interaction-based usability evaluation requires additional preprocessing steps: (1) structure-based clustering of webpages; (2) determining user intention, e.g., following the approach proposed by [8]; and (3) providing a common usability model per cluster and intention.

That is, for X types of user intention, a corresponding framework would have to provide X usability models per webpage cluster (Fig. 2). For example, assume a cluster contains all blogs using the same WordPress template. Then one would have to train different models for users *just browsing around* and users looking for a *certain piece of information* since these two intentions cause considerable differences in behavior. In the remainder of this paper, we address how to derive appropriate usability models and heuristics w.r.t. the requirements just described.

3 Usability-based Split Testing

We propose *Usability-based Split Testing*, which is a feasible trade-off between effectiveness and efficiency, as an alternative to established approaches (Fig. 1). That is, we aim at significantly better predictions of usability than can be done using conversions. Besides, we want to be more efficient than established methods of usability evaluation. To achieve this, we have designed a two-step process:

1. Track user behavior on the interfaces of a split test—i.e., the *interfaces-under-test*—and apply the resulting interaction metrics to **heuristic rules** for usability evaluation, e.g., "a higher cursor speed indicates more confusion". Test whether the difference between the interfaces is *significant.*
2. If the result is not significant, more specific information is required. Thus, add a *usability questionnaire* to one of the interfaces. From the answers and the tracked interactions, learn more specific **usability models** that can be applied to the remaining interfaces for predictions of usability.

For realizing these steps and meeting requirements **(R1)**–**(R3)**, as described in Section 1, our novel approach follows a set of well-defined principles that will be introduced in the following.

3.1 Component-based Interaction Tracking

The major goal of our approach is to overcome the problems regarding interactions and low-level page structure, as described in Section 2.3. During the study motivating this paper, interaction feature values were calculated on a very fine-grained basis. Particularly, interactions on any text or media element were considered for analysis, no matter how tiny or unimportant. This means that removing some minor elements from a webpage—such as text snippets in a sidebar—would already impact the values of interaction features. Also, only normalized absolute values were considered, rather than paying attention to relative distributions of interactions across the webpage.

To address this issue of interactions being highly dependent on low-level structure, we follow a *component-based approach.* For this, an interface-under-test, i.e., a single webpage, is divided into higher-level components such as the whole navigation bar rather than considering individual links. This approach partly follows the concepts of areas of interest [7] and critical components [15]. The rest of the webpage is treated as a separate, remaining component while the lower-level structure within a component is considered a *black box.* It is also possible to apply this to components in the context of other approaches, e.g., the WebComposition approach [5]. Since we intend to track interactions on a per-component basis, in this way small changes to the lower-level structure—e.g., removing minor text snippets—do not have an impact on feature values. *Usability models learned from such component-based interactions can then be applied to different webpages as long as the large-scale structure remains the same.*

3.2 Interaction-based Heuristic Rules for Usability Evaluation

Interactions tracked in the context of a usability-based split test can be easily applied to pre-defined heuristic rules. To give just one example, assume a rule stating that *higher cursor speed positively correlates with user confusion*. Then, if the users of one interface-under-test produce significantly lower cursor speeds than users of another, this is a clear indicator of less confusion. By design, this variant of our approach is as efficient as conversion-based split testing *(R2)*. That is, it can be very quickly deployed on online webpages and does not bother the user with requests for explicit feedback. Moreover, the collected interaction-based metrics are precise and easily interpretable using the given heuristic rules *(R3)*.

A drawback of this variant is the fact that the rules used need to be determined in a different setting of the same context (i.e., similar high-level structure, similar user intention) first. That is, a dedicated training phase is required, e.g., a controlled user study during which explicit usability judgments are correlated with interactions. Since the applied heuristic rules originate from a different setting, they cannot be a perfect measure of usability for the interfaces-under-test. Rather, they can only give reasonable approximations, but still provide more insights into users' actual behavior than conversions *(R1)*. However, if this approach fails to deliver significant results, one needs to obtain more precise information for predicting usability by leveraging corresponding models.

3.3 Leveraging Usability Models

The second variant of our Usability-based Split Testing approach uses models for predicting usability. For this, one interface-under-test is chosen to deliver training data. That is, users of the interface are presented with a questionnaire asking for explicit, quantitative judgments of usability. This questionnaire is based on Inuit [20], an instrument describing usability by a set of only seven items: *informativeness*, *understandability*, *confusion*, *distraction*, *readability*, *information density* and *accessibility*. In this way, the number of questions a user has to answer is kept to a minimum [20]. The items have also been specifically designed for meaningful correlation with client-side user behavior [20]. Together with the collected interactions, explicit judgments are then used for training appropriate models based on existing machine learning classifiers. Since the interfaces-under-test all feature the same high-level structure—in accordance with *component-based interaction tracking*—these models can be applied to the interactions of the remaining interfaces for predictions of usability.

This variant of our approach cannot reach the same efficiency as conversion-based split testing since parts of the users are faced with requests for explicit feedback. Yet, by design, it is more efficient than traditional methods such as remote asynchronous user studies *(R2)*. Given the minimum of two interfaces-under-test, only 50% of our users are presented with questionnaires, compared to 100% of the participants in a controlled user study. Moreover, our approach can be easily applied to online web interfaces. It does not require a cumbersome study set-up since we rely on interactions and judgments of real users only. In comparison to conversions or heuristic rules, models provide considerably more precise insights into users' behavior and its connection to usability *(R1)*. Also, questionnaires and models deliver an easily interpretable set of quantitative metrics in terms different usability aspects (*informativeness*, *understandability* etc.) for comparing interface performance *(R3)*.

4 The WaPPU Tool

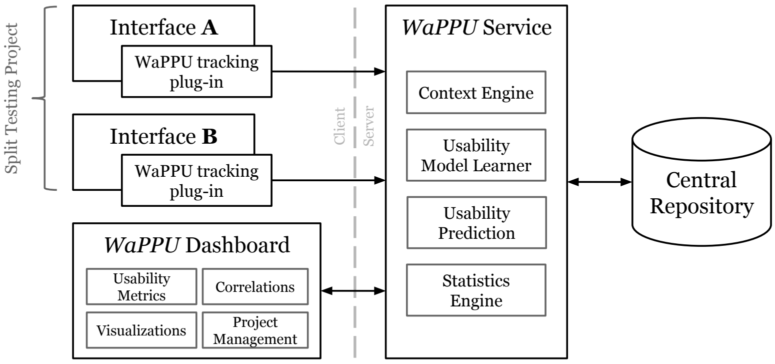


Fig. 3. Architecture of WaPPU.

To provide a ready-to-use framework for Usability-based Split Testing, we have designed a novel context-aware tool called *WaPPU*. The tool caters for the whole process from interaction tracking to deriving correlations and learning usability models. Based on the principles of Usability-based Split Testing, we have implemented WaPPU in terms of a central split testing service. This service has been realized using *node.js*[5]. Split testing projects are created in the WaPPU dashboard (Fig. 3), which provides the developer with ready-to-use JavaScript snippets that simply have to be included in the different interfaces-under-test. The only other thing required for deployment of the split test is a client-side jQuery plug-in for component-based interaction tracking. The overall architecture of WaPPU can be seen in Figure 3. The current implementation supports at most two interfaces per split test, i.e., only A/B testing is supported.

Interaction Tracking Our tool tracks a total of 27 well-defined user interaction features, of which the most expressive ones are shown in Table 1. They have been derived from existing research [7, 12, 15, 19] as well as video analyses of real search engine users. The features are tracked for each component defined by the developer, except for features annotated with an asterisk, which cannot

[5] `http://nodejs.org/` (2014-02-21).

Table 1. Selection of interaction features tracked by WaPPU (* = whole page feature only). The complete list can be found in our online appendix [22].

label	**description**	**source**
charsTyped	# characters typed	
cursorMoveTime	time the mouse cursor spends moving	[19]
cursorSpeed	*cursorTrail* divided by *cursorMoveTime*	[7, 19]
cursorSpeedX	cursor speed in X direction	[7]
cursorStops	# cursor stops	[7]
cursorTrail	length of cursor trail	[7, 19]
hovers	# hovers	[19]
hoverTime	total time spent hovering the component	[19]
pageDwellTime*	time elapsed between loading and leaving the page	[7]
scrollDirChanges*	# changes in scrolling direction	[15]
scrollMaxY*	maximum scrolling distance from top	[7]
scrollPixelAmount*	total amount of scrolling (in pixels)	[7]
textSelections	# text selections	
textSelectionLength	total length of all text selections	

be applied to individual components. Moreover, each feature is tracked for the whole web interface, which gives an additional implicit component. This gives us the chance to derive the relative distribution of features across the page, e.g., "25% of the total cursor trail lie in the navigation component". If a developer defines x components in their web interface and specifies that all features shall be considered, WaPPU tracks a total of $20(x + 1) + 7$ features during the split test (7 features are applied to the whole page instead of components).

Usability Judgments WaPPU offers the option to automatically show a questionnaire when users leave an interface-under-test, in case they have agreed to contribute training data. This questionnaire contains the seven usability items of Inuit [20], each formulated as a question and to be answered based on a 3-point Likert scale. Since the value of an item is thus either -1, 0 or $+1$, we get an overall usability value that lies between -7 and $+7$. These values are what we refer to as *quantitative measures/metrics of (aspects of) usability* in the remainder of this paper. The questionnaire can be shown on either none, one or all of the interfaces-under-test in a split test. If no interface features a questionnaire, the functionality of WaPPU is reduced to collecting interactions only, i.e., for use with *usability heuristics* (cf. Sec. 3.2).

If it is featured on one interface, WaPPU automatically learns seven models—one per usability item—based on the users' answers. These models are associated with the corresponding split testing project and stored in WaPPU's central repository (Fig. 3). They are automatically applied to the remaining interfaces for *model-based usability prediction* (cf. Sec. 3.3). The current implementation of

WaPPU uses the updateable version of the Naïve Bayes classifier[6] provided by the WEKA API [10].

Finally, in case all interfaces feature the questionnaire, the developer receives the most precise data possible. This case requires no models and is particularly useful for *remote asynchronous user studies* from which one can derive heuristic rules for usability evaluation (cf. Sec. 3.2). It is not intended for evaluation of online interfaces since the amount of questionnaires shown to real users should be minimized.

Context-Awareness The context of a user is automatically determined by WaPPU and all collected interactions and usability judgments are annotated accordingly. In this way, it is possible to integrate context into a usability model since different contexts trigger different user behaviors. Currently, we consider two aspects that to a high degree influence a user's interactions: *ad blockers* and *screen size*. That is, the context determined by our tool is a tuple $(\mathit{adBlock}, \mathit{screenSize})$ with $\mathit{adBlock} \in \{\mathit{true}, \mathit{false}\}$ and $\mathit{screenSize} \in \{\mathit{small}, \mathit{standard}, \mathit{HD}, \mathit{fullHD}\}$. For this, we refer to the most common screen sizes and define: $\mathit{small} < 1024 \times 768 \leq \mathit{standard} < 1360 \times 768 \leq \mathit{HD} < 1920 \times 1080 \leq \mathit{fullHD}$.[7]

Small-screen and touch devices are not supported in the current version of WaPPU. They are detected using the MobileESP library[8] and corresponding data are ignored.

5 Evaluation

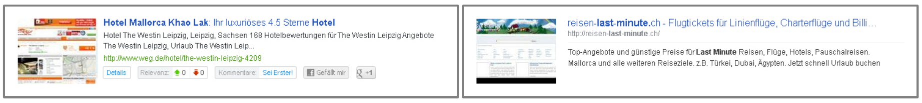
We have engaged the novel concept of Usability-based Split Testing for evaluating the interface of a real-world web search, which is currently a closed beta version and developed by the R&D department of *Unister GmbH*. For this, we have redesigned the standard search engine results page (SERP) and put both the old and redesigned versions of the interface into an A/B test using WaPPU. According to component-based interaction tracking as one principle of Usability-based Split Testing (cf. Sec. 3.1), we have defined two high-level components within the SERPs: the container element containing all search results (`#serpResults`) and the container element containing the search box (`#searchForm`). From this split test, we have obtained (a) an evaluation of the two SERPs, (b) corresponding usability models and (c) a set of interaction-based heuristics for general use with SERPs of the same high-level structure. Results suggest that our approach can effectively detect differences in interface usability and train models of reasonable quality despite a rather limited set of data.

The following describes the research method for evaluating the approach, the concrete test scenario, and presents the evaluation results. Datasets for reproducing results and detailed figures can be found in our online appendix [22].

[6] `http://weka.sourceforge.net/doc.dev/weka/classifiers/bayes/NaiveBayesUpdateable.html` (2013-10-07).

[7] Cf. `http://en.wikipedia.org/wiki/Display_resolution` (2014-02-12).

[8] `http://blog.mobileesp.com/` (2014-02-12).



Hotel Mallorca Khao Lak: Ihr luxuriöses 4.5 Sterne **Hotel**
Hotel The Westin Leipzig, Leipzig, Sachsen 168 Hotelbewertungen für The Westin Leipzig Angebote The Westin Leipzig, Urlaub The Westin Leip...
http://www.weg.de/hotel/the-westin-leipzig-4209

reisen-**last-minute**.ch - Flugtickets für Linienflüge, Charterflüge und Billi...
http://reisen-**last-minute**.ch/

Top-Angebote und günstige Preise für **Last Minute** Reisen, Flüge, Hotels, Pauschalreisen. Mallorca und alle weiteren Reiseziele. z.B. Türkei, Dubai, Ägypten. Jetzt schnell Urlaub buchen

Fig. 4. Search result from the original SERP (left) vs. search result from the novel SERP redesigned by three usability experts (right).

5.1 Method

The evaluation was carried out as a remote asynchronous user study whose workflow oriented at [13]. Participants were recruited via internal mailing lists of the cooperating company. Since user intention considerably affects interactions (cf. Sec. 2.3), we intended to minimze fluctuations in this respect. Thus, we definined a *semi-structured* task to simulate that all participants act according to a common intention, i.e., "Find a birthday present for a good friend that does not cost more than 50 Euros." We assumed that the vast majority of users would not immediately have an adequate present in mind and thus behave like *information gatherers* [8]. Additionally, in order to reduce context to different screen sizes only, participants were instructed to disable any ad blockers.

Each participant was randomly presented with one of the two SERP interfaces for completing their task. Before leaving a results page, they had to rate its usability using the Inuit questionnaire displayed by WaPPU. That is, we used a configuration of WaPPU which triggers a questionnaire in both interfaces in the A/B test. Since a user might trigger several searches and thus view several different SERPs, they potentially had to answer the questionnaire more than one time during the study. This means that one study participant could produce several datasets, each containing interactions and usability judgments. Answering one questionnaire per results page is necessary since different searches lead to different results, which influences usability items such as *informativeness* and *information density*. Participants were instructed to click a "finish study" button, once they found an adequate present. Clicking the button redirected to a final questionnaire asking for demographic information.

5.2 Interface Redesign

The web search's standard SERP interface was redesigned by three experts in order to increase its usability. The redesign was carried out according to established guidelines as well as the experts' own experiences with interface design and usability evaluations. One of the experts was a graphic designer and one was an interaction designer holding an according Master's degree, both with several years of experience in industry. The third expert was a PhD student with focus on human-computer interaction.

Some representative points concerning the redesign were: (a) better visual separation of results, (b) more whitespace, (c) giving text more space in terms of a greater line height, (d) aligning text and image elements more consistently,

(e) removing unnecessary comment and social media buttons and (f) reducing the amount of advertisements. Although the changes were rather subtle, we particularly assumed less *confusion* and *distraction* as well as better *readability* and *information density*. A visual comparison of exemplary search results from the two interfaces can be found in Fig. 4.

5.3 Results

We recruited 81 unique participants who contributed 198 individual datasets, i.e., they triggered 198 searches/SERPs. 17 of the participants were familiar with the investigated web search (i.e., they had used it before); 37 answered the final questionnaire (23 male). In general, participants stated they privately surf the internet for 2–3 hours per day, mostly for social networking (N=23) and reading news (N=22). The mostly used search engine is Google (N=35), in general several times a day (N=34) and usually for knowledge (N=31) and product search (N=21). On average, participants were 31.08 years old (σ=5.28).

During the study, we registered two different contexts: *HD* (N=46) and *full HD* (N=35). One participant was excluded from the analysis, because they delivered invalid data. For our evaluation, we additonally distinguish between users who were *not familiar* and users who were *familiar* with the web search since they produced considerably different results.

Interface Evaluation First, we have a look at the usability evaluations of the two SERP interfaces w.r.t. the questionnaires filled out by the participants. That is, we investigate whether our approach was able to detect the difference in usability originating from the experts' interface redesign. For this, we have carried out four analyses regarding the two contexts *HD* and *full HD* as well as familiarity with the investigated web search.

The largest amount of datasets (89) was produced by users with HD screens who were not familiar with the web search. Participants also not familiar with the interface, but using full HD screens contributed 52 datasets. This makes a total of 141 datasets from novel users. In contrast, participants who were familiar with the web search contributed 47 datasets (30 HD, 17 full HD).

As is apparent from Table 2, the largest group of users (*HD*/*not familiar*) found the redesigned SERP to be significantly[9] better regarding the aggregated usability (i.e., the sum of all individual items; μ=2.45, σ=2.46). Moreover, the new interface performed significantly better concerning *distraction* (μ=0.62, σ=0.62) and *information density* (μ=0.43, σ=0.67). This matches our assumptions based on the experts' redesign. In general, the new SERP performs better regarding all usability items, although not always statistically significant.

In contrast, analysis of HD screen users familiar with the web search did not show a significant overall difference between the two SERPs. Yet, they judged the old interface to be significantly better concerning the individual item *confusion*

[9] All tests of significance were carried out using the *Mann–Whitney U test* (α=0.05), which is particularly suitable for non-normally distributed independent samples.

Table 2. Evaluations by participants not familiar with the web search who used an *HD* screen (A = old interface, B = new interface).

	A (N=47)		B (N=42)		
usability item	μ	σ	μ	σ	significance
informativeness	-0.17	0.84	-0.02	0.84	—
understandability	0.34	0.70	0.45	0.67	—
confusion	0.30	0.78	0.38	0.70	—
distraction	0.36	0.74	0.62	0.62	p<0.05, W=798.5
readability	0.45	0.65	0.52	0.71	—
information density	0.04	0.69	0.43	0.67	p<0.01, W=692
accessibility	0.06	0.67	0.07	0.75	—
usability	1.38	2.96	2.45	2.46	p<0.05, W=782

(μ=0.69, σ=0.48). On average, it was also judged to be less *distracting* (μ=0.85, σ=0.38) and have better *readability* (μ=0.62, σ=0.51) and *accessibility* (μ=0.38, σ=0.65). This finding is contrary to our assumptions. Rather, it indicates that users get accustomed to suboptimal interfaces and seem to be confused by changes even if they yield better usability from a more objective point of view.

Concerning the context *full HD/not familiar*, our analysis shows no significant differences between the two interfaces. However, results suggest that the usability of the old interface is better on average. Contrary, the redesigned SERP on average indicates better performance regarding *information density* (μ=0.19, σ=0.83) and *confusion* (μ=0.06, σ=0.85). Finally, full HD users who were familiar with the web search saw the biggest difference between the two interfaces. They judged the new SERP to be significantly better concerning *distraction* (μ=0.80, σ=0.42), *readability* (μ=0.60, σ=0.52), *information density* (μ=0.10, σ=0.57), *accessibility* (μ=0.50, σ=0.71) and aggregated usability (μ=2.60, σ=2.41). However, this context contained the smallest number of datasets (17) and therefore cannot be considered to be representative.

Usability Models Based on the most representative dataset (Tab. 2) we have trained and tested Random Forest[10] classifiers for predicting usability across interfaces. This is in analogy to WaPPU's functionality of providing the questionnaire only in one interface and guessing the usability of the second interface from automatically learned models. Particularly, we intend to investigate whether component-based interaction tracking (cf. Sec. 3.1) is feasible for predicting the usability of a different webpage that did not contribute training data. For this, we take interaction data and usability judgments from the old SERP and train models from these—one for each Inuit usability item. The interaction data from the redesigned SERP are used as the test set for these models.

In a first step, we have selected the most expressive interaction features for each model. This has been done using *Correlation-based Feature Subset Selection* [9] in combination with best-first search. That is, we have selected "[s]ubsets of

[10] `http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/RandomForest.html` (2014-03-23).

features that are highly correlated with the class [to be predicted] while having low intercorrelation"[11]. Both functions are provided by the WEKA API [10]. Subsequently, we have trained the models based on the selected features and used our test set to evaluate them.

In general, the quality of the trained models was reasonably good. We obtained the most precise predictions for the item *distraction* (F-measure = 0.518), which was also one of the significant items for the considered context. In contrast, the item *readability* yielded the least precise predictions (F-measure = 0.296). The amount of training and test data was rather small for the investigated context (47 and 42 data sets, respectively). Thus, we assume better prediction quality with a larger amount of real-world users since correlations would then become more homogeneous, as has been observed in [19]. The precise results of the model evaluation can be found in our online appendix [22].

6 Key Findings of the User Study

The results from the largest and most representative group of participants *HD/not familiar* (Tab. 2) confirm that our approach is able to effectively detect differences in the usability of two versions of the same interface. We moreover found that users get used to interfaces and thus become less receptive to adjustments, even if these aim at better usability. What remains to be investigated are the differences between judgments from HD and full HD users. In particular, it requires deeper insights into users' actual behavior to understand why users familiar with the investigated web search produced very contradictory evaluations when differentiating between screen resolutions.

The usability models trained from our data underpin that the component-based tracking approach can reduce variations in users' interactions, which are caused by differences in lower-level structure. This was a major problem during the motivating study (cf. Sec. 2.3). Results suggest that WaPPU is able to predict interface usability based on adequate models with reasonable effectiveness.

Based on the feature selection process for learning usability models and Pearson's correlations r, we have additionally derived heuristic rules for SERPs, which are summarized in Figure 5. Regarding the dataset these rules are based on, their validity is theoretically restricted to *HD screen* users. Still, they can be applied to any SERP—as long as it is of similar structure and has the same components defined as the SERPs investigated in our evaluation—since many of the included features (e.g., *page dwell time*) do not strongly depend on screen resolution. To give just one example, a developer could monitor interactions on two SERPs. If *page dwell time* and *maximum scrolling distance* are significantly lower on one SERP, this is a clear signal for better *information density.* Yet, results must be interpreted with caution, as we have only investigated the user type *information gatherer* in our study. If it is not possible to obtain significant

[11] `http://weka.sourceforge.net/doc.dev/weka/attributeSelection/CfsSubsetEval.html` (2014-02-20).

- Better **informativeness** is indicated by
  - a lower absolute *cursor speed* on the search box (r=-0.21);
  - a higher relative amount of *hovers* on the search results (r=0.40).
- Better **understandability** is indicated by
  - a lower absolute *cursor speed* on the search box (r=-0.46);
  - a higher relative amount of *hovers* on the search results (r=0.24).
- Less **confusion** is indicated by
  - a lower relative *cursor speed (X axis)* on the search box (r=-0.49);
  - a lower absolute *maximum scrolling distance from top* (r=-0.44);
  - a lower absolute *amount of scrolling (in pixels)* (r=-0.33).
- Less **distraction** is indicated by
  - a lower absolute amount of *cursor stops* (r=-0.26);
  - a smaller absolute *length of the cursor trail* (r=-0.25).
- Better **readability** is indicated by
  - a lower absolute *page dwell time* (r=-0.21);
  - a smaller absolute amount of *text selections* (r=-0.27);
  - a smaller absolute *length of text selections* (r=-0.39).
- Better **information density** is indicated by
  - a lower absolute *page dwell time* (r=-0.11);
  - a lower absolute *maximum scrolling distance from top* (r=-0.27).
- Better **accessibility** is indicated by
  - a lower absolute amount of *characters typed* into the search box (r=-0.27);
  - a lower absolute amount of *changes in scrolling direction* (r=-0.31).

Fig. 5. Heuristic rules for usability evaluation of SERPs, as derived from our user study.

results from the heuristics, one must switch to a more effective method—e.g., leveraging specifically trained models, as described earlier.

7 Limitations and Future Work

We are aware of the fact that usability is a hard-to-grasp concept that is difficult to measure in an objective manner—if possible at all. However, our approach is able to yield reasonable approximations of usability in a quantitative and easy-to-understand form. This is particularly valuable in today's IT industry with its short time-to-market. If existing conversion-based analyses are augmented with Usability-based Split Testing, it will be possible to detect major shortcomings in web interfaces without having to carry out costly and/or time-consuming evaluations (yet, our approach only detects differences between interfaces-under-test and does not directly drive adequate changes for better usability). If results delivered by our method are not significant, it is still possible to apply such evaluations, which are more effective yet less efficient. However, we intend to minimize the need for the latter.

As has been pointed out earlier (cf. Sec. 2.3), a user's intention has considerable impact on their behavior. However, we have not yet considered intention as a factor in the design of our Usability-based Split Testing tool WaPPU. Rather, we modeled the user study for our evaluation in such a way that all users had to behave the same. Currently, we are investigating how intention can be derived from the interaction features tracked by WaPPU. According to [8], features such as the *page dwell time* can indicate user behavior. In future versions of WaPPU, we intend to add an extra question to the questionnaire asking for the user's intention. In this way, we can train an additional model for determining intention before applying adequate usability models or heuristics.

The current version of WaPPU is restricted to processing mouse and keyboard input. Yet, small-screen touch devices are gaining more and more popularity. Therefore, a major part of our future work will be to transfer Usability-based Split Testing into the context of touch devices. It will be particularly interesting to investigate how the different set of interaction features (e.g., missing *cursor trail*, new *zooming interaction*) affects usability prediction quality. First steps into this direction have already been taken by Nebeling et al. [15].

Finally, we intend to integrate our approach into the WebComposition process model [5] for enabling continuous evaluation of evolving widget-based interfaces.

8 Conclusions

This paper has presented *Usability-based Split Testing*—a novel method for *ensuring web inferface quality* based on quantitative metrics and user interactions. We have also introduced a corresponding A/B testing tool called *WaPPU*. Our approach intends to determine the usability of an interface more effectively than conversion-based methods while being more efficient than traditional approaches like expert inspections or controlled lab studies. To realize this, our method determines usability based on users' interactions. That is, we track interactions and apply them to either pre-defined heuristic rules or models trained with the help of users who answered an additional questionnaire. In this way, we obtain quantitative approximations of usability for empirically comparing web interfaces.

In a user study with 81 participants, we have applied our approach to the standard version and a redesigned version of a real-world SERP. Results show that our tool is able to detect the predicted differences in usability at a statistically significant level. Moreover, we were able to train usability models with reasonable prediction quality. Additionally, a set of key usability heuristics for SERPs could be derived based on user interactions. The study findings underpin the feasibility of the proposed approach.

Future work includes transferring the approach into the context of touch devices. Moreover, future versions of WaPPU shall be able to determine a user's intention before selecting appropriate usability heuristics and models.

Acknowledgments We thank Robert Frankenstein, Björn Freiberg, Viet Nguyen, Thomas Stangneth and Katja Zatinschikow for helping with design and implementation of the user study. This work has been supported by the ESF and the Free State of Saxony.

Gefördert aus Mitteln der Europäischen Union

Europa fördert Sachsen. ESF Europäischer Sozialfonds

References

1. Atterer, R., Wnuk, M., Schmidt, A.: Knowing the Users Every Move – User Activity Tracking for Website Usability Evaluation and Implicit Interaction. In: Proc. WWW. (2006)
2. Carta, T., Patern, F., Santana, V.F.: Web Usability Probe: A Tool for Supporting Remote Usability Evaluation of Web Sites. In: Campos, P., Graham, N., Jorge, J., Nunes, N., Palanque, P., Winckler, M. (eds.) INTERACT 2011. LNCS, vol. 6949, pp. 349–357. Springer, Heidelberg (2011)
3. Correani, F., Leporini, B., Patern, F.: Automatic Inspection-based Support for Obtaining Usable Web Sites for Vision-Impaired Users. UAIS 5(1) (2006)
4. de Vasconcelos, L.G., Baldochi Jr., L.A.: Towards an Automatic Evaluation of Web Applications. In: Proc. SAC. (2012)
5. Gaedke, M., Gräf, G.: Development and Evolution of Web-Applications using the WebComposition Process Model. In: WWW9-WebE Workshop. Amsterdam (2000)
6. Gay, G.R., Li, C.Q.: AChecker: Open, Interactive, Customizable, Web Accessibility Checking. In: Proc. W4A. (2010)
7. Guo, Q., Agichtein, E.: Beyond Dwell Time: Estimating Document Relevance from Cursor Movements and other Post-click Searcher Behavior. In: Proc. WWW. (2012)
8. Gutschmidt, A.: Classification of User Tasks by the User Behavior. PhD thesis, University of Rostock (2012)
9. Hall, M.A.: Correlation-based Feature Subset Selection for Machine Learning. PhD thesis, University of Waikato (1998)
10. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. SIGKDD Explor. Newsl. 11(1) (2009)
11. Hassenzahl, M.: Hedonic, emotional and experiential perspectives on product quality. In: Ghaoui, C. (ed.) Encyclopedia of Human Computer Interaction, pp. 266-272. IGI Global (2006)
12. Navalpakkam, V., Churchill, E.F.: Mouse Tracking: Measuring and Predicting Users' Experience of Web-based Content. In: Proc. CHI. (2012)
13. Nebeling, M., Speicher, M., Norrie, M.C.: CrowdStudy: General Toolkit for Crowdsourced Evaluation of Web Interfaces. In: Proc. EICS. (2013)
14. Nebeling, M., Matulic, F., Norrie, M.C.: Metrics for the Evaluation of News Site Content Layout in Large-Screen Contexts. In: Proc. CHI. (2011)
15. Nebeling, M., Speicher, M., Norrie, M.C.: W3Touch: Metrics-based Web Page Adaptation for Touch. In: Proc. CHI. (2013)
16. Nielsen, J., Molich, R.: Heuristic Evaluation of User Interfaces. In: Proc. CHI. (1990)
17. Nielsen, J.: Putting A/B Testing in Its Place, `http://www.nngroup.com/articles/putting-ab-testing-in-its-place/`
18. Sauro, J.: Does Better Usability Increase Customer Loyalty? `http://www.measuringusability.com/usability-loyalty.php`
19. Speicher, M., Both, A., Gaedke, M.: TellMyRelevance! Predicting the Relevance of Web Search Results from Cursor Interactions. In: Proc. CIKM. (2013)
20. Speicher, M., Both, A., Gaedke, M.: Towards Metric-based Usability Evaluation of Online Web Interfaces. In: Mensch & Computer Workshopband. (2013)
21. Speicher, M., Both, A., Gaedke, M.: Was that Webpage Pleasant to Use? Predicting Usability Quantitatively from Interactions. In: Sheng, Q.Z., Kjeldskov, J. (eds.) Current Trends in Web Engineering. LNCS, vol. 8295, pp. 335–339. Springer, Heidelberg (2013)
22. WaPPU Online Appendix, `http://vsr.informatik.tu-chemnitz.de/demo/WaPPU`