Julius-Maximilians-Universität Würzburg
Institut für Informatik
Lehrstuhl für Kommunikationsnetze
Prof. Dr. T. Hoßfeld

Estimating Quality of Experience of Enterprise Applications – A Crowdsourcing-based Approach

Kathrin Johanna Borchert

Würzburger Beiträge zur
Leistungsbewertung Verteilter Systeme

Bericht 02/20

Würzburger Beiträge zur
Leistungsbewertung Verteilter Systeme

Herausgeber

Prof. Dr. T. Hoßfeld, Prof. Dr.-Ing. P. Tran-Gia
Universität Würzburg
Institut für Informatik
Lehrstuhl für Kommunikationsnetze
Am Hubland
D-97074 Würzburg

Tel.: +49-931-31-86631
Fax.: +49-931-31-86632
email: tobias.hossfeld@uni-wuerzburg.de

Satz

Reproduktionsfähige Vorlage des Autors.
Gesetzt in LaTeX Linux Libertine 10pt.

ISSN 1432-8801

Estimating Quality of Experience of Enterprise Applications – A Crowdsourcing-based Approach

Dissertation zur Erlangung des
naturwissenschaftlichen Doktorgrades
der Julius–Maximilians–Universität Würzburg

vorgelegt von

Kathrin Johanna Borchert

geboren in

Frankfurt am Main

Würzburg 2020

Eingereicht am: 14.08.2020
bei der Fakultät für Mathematik und Informatik
1. Gutachter: Prof. Dr. Tobias Hoßfeld
2. Gutachter: Dr. Katrien De Moor (Assoc. Prof.)
Tag der mündlichen Prüfung: 26.10.2020

Danksagung

Ich möchte die Gelegenheit nutzen und an dieser Stelle den vielen Menschen, die mich auf dem Weg zur Promotion begleitet, unterstützt, motiviert und gefördert haben, meinen Dank aussprechen.

Zu allererst möchte ich mich bei Prof. Dr. Tobias Hoßfeld und Prof. Dr.-Ing. Phuoc Tran-Gia dafür bedanken, dass sie mir die Möglichkeit zur Promotion am Lehrstuhl für Kommunikationsnetze gegeben haben. Die Zeit dort umfasste nicht nur das Mitwirken an spannenden Projekten, sondern auch viele Reisen zu Konferenzen und Workshops, um dort die in den Projekten entstandenen Forschungsergebnisse zu präsentieren und in den Austausch mit Wissenschaftlern aus der ganzen Welt zu kommen. Zusätzlich zu den Forschungsarbeiten konnte ich Einblicke in viele verschiedene Tätigkeitsfelder gewinnen, beginnend bei der Gestaltung der Lehre bis hin zur Mitwirkung an akademischen Prozessen im Rahmen von Berufungen und der Unterstützung der Studierenden bei der Planung und Umsetzung von Auslandsaufenthalten. Dies ermöglichte es mir mich fachlich und menschlich weiterzuentwickeln.

Darüber hinaus möchte ich mich insbesondere bei Prof. Dr. Tobias Hoßfeld für seine Betreuung im Rahmen des Promotionsprozesses bedanken. In diesem Zusammenhang geht auch ein großes Dankeschön an Assoc. Prof. Dr. Katrien De Moor, welche das Zweitgutachten zu meiner Dissertationsschrift verfasst hat, sowie an Prof. Dr. Andreas Hotho und Prof. Dr. Birgit Lugrin, welche sich bereit erklärt haben Teil der Prüfungskommission meiner Disputation zu sein.

Natürlich wäre der Arbeitsalltag am Lehrstuhl nicht das gewesen was er war ohne die vielen fachlichen Diskussionen und Gespräche bei einem Kaffee oder Tee und die unterhaltsamen Aktivitäten wie Betriebsausflüge und Weihnachts-

feiern mit meinen ehemaligen Kollegen und Kolleginnen. Ein besonderer Dank geht an meinen ehemaligen Gruppenleiter Matthias Hirth. Ohne die oft diskussionsreiche Zusammenarbeit, gemeinsamen Publikationen und aufbauenden Gespräche wäre die Promotion in der Form nicht möglich gewesen. Bei Thomas Zinner möchte ich für seine große Unterstützung, die oft auch über das fachliche hinaus reichte, bedanken. Für seine konstruktive Kritik in der finalen Phase der Promotion, die maßgeblich zum Gelingen dieser beigetragen hat, danke ich Michael Seufert. Ebenfalls möchte ich Stefan Geißler und Susanna Schwarzmann, mit denen ich jeweils eine Zeit lang das Büro teilen durfte, Florian Metzger, Anika Seufert, Alexej Grigorjew, Nicholas Gray, Florian Wamser, Frank Loh, Nikolas Wehner, Christian Moldovan, Christopher Metter, Stanislav Lange, Anh Nguyen-Ngoc, Lam Dinh-Xuan, Valentin Burger, Steffen Gebert und Christian Schwartz meinen Dank aussprechen für die gemeinsame Zeit.

Im Zusammenhang mit dem Leben am Lehrstuhl darf auch Alison Wichmann nicht unerwähnt bleiben. Bei ihr möchte ich mich für die Unterstützung bei der Organisation von Dienstreisen und ihr immer offenes Ohr für mich bedanken. Ebenfalls geht ein großer Dank an die studentischen Hilfskräfte, welche tatkräftig an Projekten und Aufgaben am Lehrstuhl mitgearbeitet haben, und an die Studierenden, deren Abschlussarbeiten ich während meiner Zeit am Lehrstuhl betreut habe.

Letztlich wäre diese Arbeit aber nicht ohne den Rückhalt meiner Familie und Freunde entstanden. Meinen Eltern, Elisabeth und Werner, danke ich für die Förderung während des Studiums, den Rückhalt und ihr unerschütterliches Vertrauen darin, dass ich meinen Weg gehen werde. Für viele schöne Momente, Gespräche und Unternehmungen, die mich den doch häufigen Arbeitsstress haben vergessen lassen, danke ich meinen Geschwistern und Freunden. Abschließend möchte ich mich bei Christopher Metter für seine Geduld mit mir, seine Aufmunterungen und Motivation gerade in den letzten Monaten der Promotion bedanken.

Contents

1 Introduction **1**

1.1 Scientific Contribution 4
1.2 Outline of the Thesis 8

2 Improvement of Result Quality in Crowdsourced Tasks **11**

2.1 Background and Related Work 13
2.1.1 Influence Factors on Task Selection and Performance . 14
2.1.2 Worker-based Quality Control 17
2.1.3 Aspects of Task-based Quality Improvements 19
2.2 Using Attention Testing for Quality Assurance 22
2.2.1 Introduction to Attention Testing 22
2.2.2 Description of Attention Test attentiveWeb 25
2.2.3 User Studies on Applicability 27
2.2.4 User Studies on Predictive Validity 36
2.2.5 User Study on Filter Validity 38
2.3 Task Selection - The Workers' Perspective 40
2.3.1 Survey Description 41
2.3.2 Survey Conduction 42
2.3.3 Evaluation of Survey Results 45
2.4 Impact of Task Decomposition 54
2.4.1 Study Design and Conduction 54
2.4.2 Impact on Work Performance 57

2.5 Impact of Working and Instruction Language 62
2.5.1 Methodology 63
2.5.2 Evaluation 66
2.6 Lessons Learned 71

3 Monitoring QoE of Enterprise Applications 73

3.1 Background and Related Work 75
3.1.1 Introduction to Quality of Experience 76
3.1.2 Quality Assessment of Business Applications 78
3.1.3 Influence of Study Design on QoE 80
3.2 Concept for Monitoring and Modeling QoE in Enterprise Environments 82
3.2.1 Identification of Performance Issues 82
3.2.2 Evaluation of Influence Factors 84
3.2.3 Modeling of QoE of Affected Applications 86
3.3 Dimensions of Monitoring QoE of Business Applications 87
3.3.1 Description of Dimensions 87
3.3.2 Influence of System Artificiality 89
3.3.3 Influence of Domain Knowledge 97
3.4 Designing a Survey Tool for Quality Assessments in the Wild . 106
3.4.1 Enterprise-specific Requirements 106
3.4.2 Tool Description 108
3.4.3 Analysis of Applicability 113
3.5 Lessons Learned 119

4 Towards Estimating QoE of Enterprise Applications 121

4.1 Background and Related Work 124
4.1.1 Factors Influencing the QoE of Interactive Web Applications 124
4.1.2 QoE Modeling 127

4.2 Long-term User Study 130
4.2.1 Study Description 131
4.2.2 Data Set 135
4.3 Analysis of User Behavior for Push and Pull Systems 136
4.3.1 Analysis of Users' Motivation 136
4.3.2 Temporal Assessment of Rating Behavior 139
4.3.3 Analysis of Rating Opinions 142
4.4 QoE Modeling Based on Requested Quality Assessments (Pull Approach) 144
4.4.1 Correlation Analysis 144
4.4.2 Threshold-Based Model 153
4.5 QoE Modeling Based on Self-motivated Quality Assessments (Push Approach) 157
4.5.1 Correlation Analysis between Self-motivated Ratings and Objective Metrics 157
4.5.2 QoE Models 159
4.6 Lessons Learned 163

5 Conclusion **167**

Appendix A Summary of Applied Tests **177**

Appendix B Screenshot and Questionnaires of User Studies **181**

Appendix C Parameters contained in Monitoring Data **195**

Bibliography and References **197**

1 Introduction

The ongoing globalization and digitalization of the modern working world open new possibilities of work organization and processing. Following this trend, enterprises integrate IT solutions in their business processes to optimize workflows and increase efficiency. As a result, employees have to work with applications, technical services, and systems every day for hours. From the technical perspective these applications and services are often operated remotely in large data centers or clouds for benefits in terms of flexibility and scalability. However, performance degradation, e.g., network delays or load peaks in the data center, might be perceived negatively by the employees, increase frustration, and might also have a negative effect on their productivity. The assessment of the application's performance in order to provide a smooth operation of the application is part of the application management. Within this process it is not sufficient to assess the system performance solely on technical performance parameters, e.g., response or loading times. These values have to be set into relation to the perceived performance quality on the user's side.

A concept, which follows this strategy, is the concept of Quality of Experience (QoE). QoE describes the quality of an application, service, or system as perceived by the user [18]. By using QoE models the QoE can be predicted based on objective metrics. Due to the subjective nature of the QoE, building such models requires a ground truth in terms of the individual opinion of the users. Such quality assessments are collected in user studies. Here, the users rate the perceived application's quality under specific conditions, e.g., a certain network delay. Based on the ratings, factors influencing the perception are identified, evaluated, and considered in the QoE model. The integration of the model

in the application or network management workflow provides the opportunity to use resources efficiently while considering the user's satisfaction. Related to the context of this thesis, estimating the QoE of business applications requires also an underlying QoE model.

As the perception of humans depends on multiple factors, the evaluation of these QoE influence factors is one challenge of QoE modeling. The investigation of such influence factors for a diverse set of applications was subject of multiple studies in previous research, e.g., for multimedia services, web applications, and VoIP services. For these applications relationships between user-provided ratings and objective metrics were found and QoE models were built. However, there is little knowledge about the QoE of business applications yet. It is unclear if achieved insights, proposed monitoring standards, and models for applications with similar characteristics, e.g., web-based applications, are transferable to this domain. Especially, as there are obvious differences between enterprise applications and applications used in leisure time. To name a few, the employees have no free choice which application to use, the applications have to be used frequently and on a regular basis, and their complexity often requires domain knowledge. Further, the context of usage might be more relevant, e.g., employees might be less tolerant of performance issues while talking to customers. These differences even more point out the necessity of proper methodologies for QoE studies in the domain of business applications.

Besides the traditional approach to run subjective experiments for collecting quality ratings in labs, crowdsourcing was established as a tool for conducting studies and acquiring study participants. Crowdsourcing, a composition of the terms *crowd* and *outsourcing*, was popularized by Jeff Howe in 2006 [19]. He described it as the process of outsourcing a job traditionally performed by an employee to a large and unknown group of people via an open call. Since crowdsourcing covers a wide range of diverse activities and encompasses many different practices, a variety of definitions exists nowadays [20].

Crowdsourcing-based user studies often focus on a specific form of crowdsourcing – the *microtasking*. Here, small, simple, and repetitive tasks are solved

by the *crowd* (*workers*) typically for a small financial reward [21]. Common microtasks are, for example, data engineering jobs, e.g., enhancement and verification of given data sets, and testing or research tasks ranging from user studies to software application testing. The completion of microtasks is typically requested by companies, researchers, or private persons, the commonly called *employers* or *requesters*. Traditionally, the relationship between employers and crowd workers is anonymous and there is no direct communication between these parties. Therefore, *crowdsourcing platforms* act as a mediator. Examples for popular microtasking platforms are Amazon Mechanical Turk (MTurk)[1] and Microworkers[2]. Another platform, often used in the crowdsourcing research community, was Figure Eight[3] (formerly known as CrowdFlower). This platform was a meta-platform, meaning that most of the workers were recruited from other microtasking platforms.

By using these platforms, user studies can be conducted online and unsupervised. The unsupervised condition, in combination with the anonymity and temporal nature of the employer-worker-relationship, lead to numerous challenges and research questions. One of the main issues is the assurance of the quality of task and test results. Due to missing communication between employers and workers, the task design including the description of the requested work is an essential factor. Ambiguous instructions may lead to misunderstandings about the requested work and result in unexpected, unsatisfied outcomes. Other reasons may be unreliable workers who are solely focused on the maximization of their gains while minimizing their efforts. To solve this problem, in the past years multiple quality control mechanisms were introduced and best practices for crowd-based QoE studies were published [22]. However, most of the approaches to improve the result quality focus on the employer's perspective while demands and preferences of the workers faded into the background or were even not considered at all.

[1] https://www.mturk.com/; Accessed: August 1st, 2020
[2] https://www.microworkers.com/; Accessed: August 1st, 2020
[3] https://www.figure-eight.com/; Accessed: March 1st, 2020

Investigating the QoE of business applications with laypersons, e.g., acquired from a microtasking platform, might be not representative for the perception of the employees. A context-related approach, that is in focus of this monograph, is monitoring the QoE from employees during their regular work. The conduction of QoE studies with employees in enterprises brings on new research challenges. Aspects relevant to the enterprise, such as the prevention of interrupting critical working processes and cost factors, need to be considered as well as effects on the QoE need to be investigated.

This thesis contributes to solve the discussed challenges by presenting research on QoE monitoring and modeling of enterprise applications. Besides focusing on estimating QoE through QoE monitoring within the enterprise, the main challenge of crowd-based studies and tasks – the quality assurance – is addressed. The following sections provide an overview about the scientific contribution and present the outline of this thesis.

1.1 Scientific Contribution

This thesis covers topics related to the process of QoE monitoring and modeling of enterprise applications. As there is little research on the QoE of this domain yet, there are no standardized methods and strategies for the monitoring process, as available for other types of applications and services. Hence, one contribution of this work is proposing a general concept for estimating QoE in the enterprise domain. This concept includes the identification of applications affected by performance issues within the often highly complex IT landscape of the enterprise. Based on this, the evaluation of influence factors and finally, the modeling of the QoE is expounded. Besides the description and discussion of the concept, this monograph covers various aspects and open challenges of QoE estimation of enterprise applications. Here, the investigations set particular attention to the applicability of methods and models in the practice.

Figure 1.1 classifies the conducted research into the overall QoE monitoring process. The research focuses not only on QoE studies conducted in the enter-

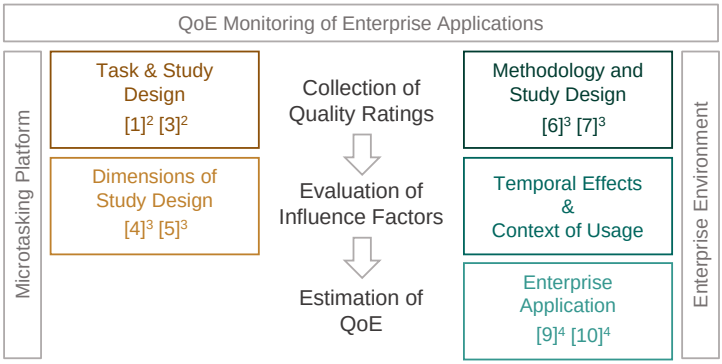


Figure 1.1: Contribution of this thesis classified into the QoE monitoring process of enterprise applications. The notation $[x]^y$ indicates that the scientific publication $[x]$ is discussed in Chapter y.

prise environment. As an established tool for subjective experiments, microtasking and its main issue, i.e., the quality assurance of task and study results, are also subject of this work. Thus, the covered research aspects are classified into research done in enterprise environments (green colored boxes) and aspects related to crowdsourcing and microtasking in particular (brown colored boxes). Further, the monitoring process is split into two parts. The first part covers methodologies for collecting quality ratings from employees in enterprise environments. This is indispensable for the QoE modeling. As the user's perception of the quality of the application might be influenced by multiple factors, the evaluation of influence factors is addressed in the second part. The evaluation comprises factors related to dimensions of the study design, e.g., the domain knowledge of the participants, and related to the context of usage. Proceeding from these steps, the focus is set to estimating QoE of enterprise applications.

Due to the subjective nature of QoE, the evaluation of influence factors on the user's perception and the creation of QoE models require the collection of quality assessments from the users. As mentioned, microtasking is a valuable strat-

egy to collect these ratings. Here, the varying quality of work and study results is one of the main issue, addressed by numerous studies. However, existing quality assurance mechanisms suffer from drawbacks such as the dependence on the task content or their expensive creation. Furthermore, approaches to improve the quality are often oriented to the needs of the employers and the worker's perspective is unattended. To overcome these limitations, this work develops a content independent quality control mechanism which identifies reliable workers based on their attention, which is a more general characteristic. This introduced filter can be applied to all types of tasks and research studies which require a certain amount of attention from the workers. Further, approaches are investigated to improve the quality of working results by optimizing the task design. The proposed approaches consider the workers' preferences when selecting and working on tasks which are derived from a survey conducted on different crowdsourcing platforms.

Instead of using microtasking for the collection of quality ratings of business applications, involving employees might lead to a more representative view on the QoE. The integration of such studies into regular working processes, which allows the consideration of contextual factors, poses new challenges. Besides requirements such as the minimization of costs, the test participation should not interrupt critical processes. These challenges are addressed by developing a non-intrusive survey tool which considers enterprise specific requirements as well as best practices for subjective experiments. The methodology is independent from a specific application and business domain and can be utilized and adapted to any interactive application with similar usage characteristics.

Regardless of whether subjective experiments are crowd-, lab-, or enterprise-based, designing them is not trivial and various aspects need to be considered as they might influence the ratings of the participants. These aspects cover characteristics related to the test participants, such as the domain knowledge, and the study environment, e.g., monitoring the QoE in the production system or in an artificial test environment. Additionally, the method used to collect quality assessments might influence the rating behavior and might result in differences

in the provided view on the QoE. A test software which uses the traditional collection method asks the users actively to rate the experienced quality after the usage of the application. A different approach, which might be easier to realize in enterprise environments, offers the users the opportunity to provide ratings on their own initiative similar to a complaint system. By discussing design dimensions and evaluating their influence on the QoE, this work sheds light on the relevance of the study design. The analysis of design aspects also focuses on the comparison of the QoE deduced from ratings acquired with the two mentioned approaches. Based on this analysis, the advantages and disadvantages of the rating systems are discussed.

With regard to the QoE estimation, another aspect which needs to be considered is changes of the application behavior over time. Changes can be, for example, caused by software updates or adjustments of the infrastructure. Especially, when modeling the QoE based on user-provided ratings and technical performance data monitored in the production system these temporal effects should be taken into account as they might affect the QoE. This topic is covered in the thesis by the exploration of influences of behavioral changes on the expectations of the employees as well as on the perception of the application's quality. Here, contextual factors such as working in different areas of a business are considered. Besides these side effects, the relationship of user-provided ratings and technical performance parameters might suffer from measurement inaccuracies due to uncontrolled conditions of real-world monitoring. Thus, it is challenging to create an accurate QoE model for enterprise applications based on noisy data, as demonstrated by previous research. To solve this issue, this work proposes two modeling approaches – threshold-based and machine learning-based. The applicability of the approaches are demonstrated on data collected in a large long-term user study conducted in a cooperating company.

To sum up, this monograph contributes to answering the following research questions.

- How to monitor the QoE of business applications in general?

- If crowdsourcing is used for subjective studies, how to optimize the quality of results by means of the task design?
- Which design dimensions need to be considered when conducting user studies with employees in enterprises?
- How to collect quality assessments from employees during their regular work?
- Does the collection method lead to differences in the provided view on the QoE?
- Do temporal effects and measurement inaccuracies influence the relations between subjective ratings and objective metrics?
- How to estimate the QoE of enterprise applications from noisy data monitored in the wild?

1.2 Outline of the Thesis

The organization of this monograph is illustrated in Figure 1.2. After this introductory chapter, the research questions are addressed in three chapters. At the beginning of each chapter background information and related work is given on the topic covered by the respective chapter. Then the studies and results are presented. In the figure research on microtasking and crowd-based approaches are colored in brown and content solely focusing on the enterprise domain is depicted in green. Each chapter summarizes the lessons learned in a concluding section.

Chapter 2 focuses on the quality improvement of the work results of crowdsourced tasks. Starting with the employer's view, in Section 2.2 a new filtering mechanism is developed to identify reliable workers based on their attention and its applicability is evaluated. Moving on to the workers' perspective, their preferences and demands on the task design are explored in Section 2.3. Based on the results of this study, two approaches to improve the performance quality of the workers are evaluated. Section 2.4 presents results on the analysis of the

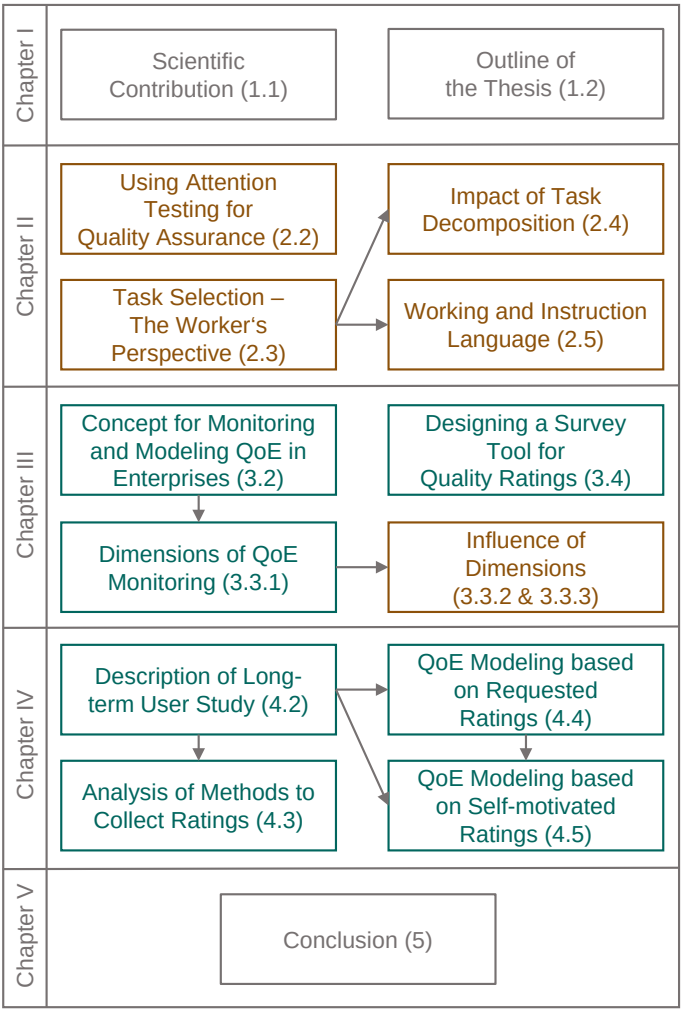


Figure 1.2: Organization of this monograph.

impact of reducing the task complexity by decomposing complex tasks into simpler sub-tasks. The second improvement method focuses on lowering language barriers by enabling the workers to process tasks in their native language. This approach is evaluated in Section 2.5.

In Chapter 3 QoE monitoring in enterprise environments is tackled. A general concept for monitoring and modeling QoE in the enterprise domain is proposed in Section 3.2. By focusing on the collection of quality assessments of enterprise applications, Section 3.3 discusses relevant dimensions of the design of QoE studies. Further, investigations on the influence of two highly relevant dimensions in the context of business applications on the QoE ratings are presented. These dimensions are the artificiality of the test system (Section 3.3.2) and the domain knowledge of the study participants (Section 3.3.3). Based on the study results as an indicator to collect quality assessments from the affected people, namely the employees, a survey tool for collecting such ratings is developed in Section 3.4. This tool allows to gather ratings from employees during their regular work. Besides the description of the used methodology, the applicability of the tool is demonstrated based on two user studies in a cooperating company.

Chapter 4 covers research towards estimating QoE of business applications. The research is based on data collected in a long-term user study with a study period of 1.5 years, as described in Section 4.2. To gain knowledge about additional influence factors on the quality ratings, Section 4.3 explores the influence of the method used to collect user-provided ratings on the QoE. Then Section 4.4 analyzes the relationship between technical performance parameters and the ratings collected with the traditional approach under consideration of temporal side effects. Based on these results, a threshold-based model for QoE estimation is introduced and its performance evaluated. Section 4.5 focuses on the relation between self-motivated ratings and objective performance metrics. By leveraging correlations, a machine learning-based QoE model is presented and compared to the threshold-based approach.

Finally, Chapter 5 concludes this monograph.

2 Improvement of Result Quality in Crowdsourced Tasks

Even if the capabilities, resources, and performance of modern computers have grown over the last decade, there are still numerous problems and tasks which are not efficiently solvable by machines. Instead, such tasks, e.g., creating ground truth data to train machine learning algorithms, writing or summarizing articles, or still tagging images, need to be solved by humans. Furthermore, several research questions need to be addressed by directly involving humans, e.g., in subjective studies.

In this context, crowdsourcing (microtasking) gives access to a large and diverse group of people. Even if this offers many opportunities, it also raises many challenges. Besides ethical challenges, e.g., unfair payments below country-specific minimum wages or the uncertainty due to missing employment contracts, requirements concerning the publishing and processing of sensitive data, and the quality assurance of the work result need to be taken into account. In this work, the quality of work results describes the degree of satisfaction of the requester with the submitted results of the workers. Thus, high quality means that the results entirely fulfill the expectation of the requester. Contrary, results of low quality do not meet the expectations at all. A commonly seen phenomenon is a high variation of the quality which may be caused by diverse reasons and influence factors. Besides worker-related reasons, e.g., unreliable or malicious workers [23], that can be overcome by including quality assurance mechanisms into the task workflow, another main influence factor is the design of the tasks including the general complexity, monetary reward, the interface design as well

Section	Research question	Methodology
2.2	Is task-independent filtering applicable and valid?	Comparison of the attention of users from an online panel and two crowdsourcing platforms using a newly developed attention test
2.3	Which design preferences have workers when selecting and working on tasks?	Analysis and comparison of answers given to a survey conducted on two crowdsourcing platforms
2.4	Which impact has task decomposition on the quality of work results?	Comparison of quality of work results of a task with and without decomposition submitted by workers from Microworkers
2.5	Does lowering language barriers improve the quality of work results?	Comparison of quality of work results produced in English or native language by workers from Microworkers

Figure 2.1: Overview about addressed research questions and used methodology.

as the instructions. A badly designed task may lead to misunderstandings concerning the expected work on the worker's side and finally results in an outcome that is unsatisfactory for the employer and the worker. Further, the design may also impact the task selection of the workers which in turn affects the overall completion time of tasks. Often, it is important for the employers to get the results as fast as possible. Hence, both parties benefit from a well designed task. However, when optimizing the task design both points of view need to be taken into account. While employers mainly focus on cost and time efficiency and a high quality of work results, the workers' perspective is more complex as aspects like intrinsic and extrinsic motivation play a major role [24].

From the viewpoint of employers and workers several research questions arise concerning the optimization of the task design. Figure 2.1 presents an overview about the research questions addressed in this chapter as well as the used methodology. Starting from the employer's point of view, a quality control mechanism, which identifies qualified and reliable workers on general characteristics and skills, is presented and evaluated. By investigating aspects of the

task design that are important for the workers while selecting tasks and working on them, the worker's perspective on a good task design is considered. Further, two strategies to reduce the complexity of tasks, i.e., by decomposing tasks into less complex sub-tasks and by providing instructions in the native language of the workers, are evaluated.

This chapter is structured as follows. Section 2.1 discusses influence factors on the task selection of workers and on the quality of work results. Further, it gives a brief overview about quality control mechanisms and related work on design optimization. The section clearly shows the gap in literature leading to the addressed research questions. Section 2.2 introduces and evaluates a new control mechanism based on the attention of the workers, mainly based on [1]. On the basis of [3], Section 2.3 sheds light on important task properties while selecting and processing tasks from the worker's point of view. Further, design preferences of workers from different platforms are analyzed and compared. Based on the results of this study, Section 2.4 evaluates how to reduce task complexity by decomposing tasks into sub-tasks, which are less complex to solve, and Section 2.5 investigates the impact of the instruction and working language – typically English – as well as an improvement by providing instructions in native language. Section 2.6 concludes this chapter.

2.1 Background and Related Work

While crowdsourcing platforms offer several advantages for the employer or researcher, such as access to a large pool of diverse workers or participants for studies [25], due to the unsupervised and uncontrolled environment several challenges must be met [26]. Besides ethical and privacy related aspects, the involvement and interaction of human beings in such an uncontrolled environment is a huge challenge. Multiple factors may influence the workers' motivation and performance on tasks. Therefore, ensuring a high quality of working results is one of the main challenges. Nowadays, explicit and implicit approaches for improving and ensuring the quality exist. Explicit approaches comprise qual-

ity assurance mechanisms which are mostly worker-related. This means, these mechanisms focus on the identification of reliable workers or their results, as well as finding workers whose skills and characteristics meet the task requirements. In contrast, implicit approaches are related to task properties, e.g. reducing the complexity or improving ambiguous instructions to avoid misunderstanding. Thus, explicit mechanisms are more suited to support the employers' search for reliable workers, while implicit approaches help reliable workers to successfully complete tasks. In this section, first a brief overview about influence factors on the task performance, and thus, on the quality of work results is given. Second, explicit and implicit approaches for improving the quality are presented and discussed.

2.1.1 Influence Factors on Task Selection and Performance

When it comes to the participation and work performance of workers in crowdsourcing tasks, multiple aspects have proven to have a significant impact. Besides intrinsic motivational factors like enjoyment, challenge, and competition, extrinsic factors set by the requester, e.g., payment or time needed for task completion, play an important role [24, 27]. These factors are individually weighted during the task selection process [28]. Considering all these factors while designing tasks is nearly impossible, especially as the factors depend on additional task characteristics specified by the task type, e.g., surveys or data engineering tasks. Thus, in this section only a brief overview about the most important factors related to the task design, i.e., payment, instructions, and complexity-related facets, is presented [29].

Typically, on microtasking platforms, the reward is defined by the employer without strict specifications from the platform providers. Choosing an appropriate salary is difficult, as it does not solely depend on the kind of task, its complexity, and required completion time, it also has a direct impact on the task selection by the workers. Different payment schemes, e.g., fixed, lottery or charity

rewards, incentivize workers with different characteristics and attitudes which may influence the outcome and the uptake time of tasks [30]. For example, tasks with higher rewards are more attractive and accordingly, these tasks are selected more frequently. Hence, paying higher rewards leads to faster, but not necessarily better results [31]. Other payment strategies, such as the confidence-based mechanism by Shah and Zhou [32], may have a positive effect on the quality of the work results. Furthermore, the rating behavior, meaning the fairness for judging the submitted results and paying the workers, also influences the task selection of the crowd. Workers try to reduce their risk of getting no reward due to unfair employer ratings while searching for tasks [33].

A design aspect that does not solely influence the task selection but also the work performance is the quality of the task description. Understanding the required work is essential to complete tasks successfully. Ambiguous instructions lead to additional time workers have to invest to understand the expected work to submit. Additionally, English is the commonly used language for describing and processing tasks on international microtasking platforms such as Amazon Mechanical Turk (MTurk) and Microworkers. Hence, language barriers may lead to misunderstandings concerning the required work as English is a foreign language for many workers [34, 35]. Gadiraju et al. [36] showed the necessity of considering the ambiguity of instructions due to the regular confrontation of workers with unclear formulations. Contrary to these findings are the results of the study from Wu and Quinn [37] about the perception of the task instruction quality by workers. Here, the quality of collected instructions of tasks from MTurk was positively rated in most of the cases. This opposing observation may be biased by the study design, e.g., the instruction sample and the small number of ratings per instruction. However, the authors found evidence about a positive effect of following best practice guidelines when creating task instructions and the selection of the tasks as well as the quality of work results. Relations between other characteristics of the instruction such as the description clarity and other task properties, like task complexity were analyzed by Gadiraju et al. [36]. However, a direct correlation between the clarity and the complexity was not found.

Nevertheless, task complexity has a significant impact on the performance of the workers [38]. Task complexity comprises various facets, e.g., the effort to achieve the specified goal (subjective complexity) or structural complexity. In this context, Cheng et al. [39] investigated the relationship between quality of work results and the length of tasks. They showed that smaller tasks (sub-tasks), created by the decomposition of larger tasks, result in higher quality. However, the maximal decomposition of tasks does not necessarily lead to higher-quality results [40].

Acknowledging the importance of the overall complexity for characterizing tasks performed by humans through computers, Yang et al. [41] worked towards gaining a deeper insight into the distribution of task complexity among crowdsourcing tasks as well as its perception among workers. As a result, the authors showed that, apart from the semantic content of a task and the used language, the features related to its visual appearance influence the perception of its complexity, and thus, can be utilized to accurately predict and measure it.

Besides the discussed, mainly extrinsic factors, it is also important to understand reasons beyond poorly designed tasks which affect the quality of the results. These reasons are related to the intention of participation of the workers. Based on the intention, two groups of workers are identifiable - workers who do not intentionally produce poor results and those who intentionally cause harm to the employers. The first group includes workers who are inattentive due to distractions or other reasons [42] and workers who act to the best of their knowledge, but nevertheless produce low-quality results, for example, due to a lack of experience with the task subject. Thus, the quality of work results may be affected by the fact that some workers will be more suitable for a given task than others, depending on the respective task requirements [43]. The second group consists of unreliable, malicious, and careless workers who generally lack the motivation to conscientiously address the task in depth but aim for minimizing their time spent and maximizing their rate wage [23]. Improving the task design would not increase the results submitted by these workers. Hence, employers need to include additional control mechanisms for ensuring the quality.

2.1.2 Worker-based Quality Control

Multiple approaches exist to identify unreliable workers or low-quality results submitted by them. These approaches are classifiable into two groups – a priori and a posteriori mechanisms [44]. As its name implies, a priori mechanisms are applied to limit the task access to a certain group of workers while a posterior mechanisms are used to assess the validity of results once they have been submitted by the workers.

A widely used a priori technique is filtering of the workers based on gold standards [45], other qualification tests, or workers' reputation on the platform. Gold standards are questions or tests for which the optimal outcome is known. In a gold standard transcription task, for example, an expert has already transcribed a hand-written text. The workers' results are compared to the gold standard, and a worker is considered qualified if her results match the gold standard. While these questions or tests by their nature are not necessarily task-specific, in practical reality, they often are. If gold standard qualification tests are to be task-specific, they have the drawback that the gold standard data must be generated for each task individually. Furthermore, a gold standard test is not suitable for all types of tasks, especially tasks with no prior known outcome, such as surveys [23]. Another downside is that workers can circumvent certain gold standard tests [46]. Likewise, workers' reputation scores on the microtasking platform are sometimes calculated based on having administered gold standard tests. Thus, the drawbacks of gold standard tests, at least in part, equally apply to the reputation score.

A posteriori mechanisms are mostly realized by integrating them into the task workflow. Examples are the assessment of results from one worker based on her answers given to so-called consistency questions or from multiple workers through agreement or majority votes. Consistency checks describe strategies where slightly rephrased questions are repeated during the task or incompatibilities in answers are identified, e.g., a worker who first claims to be 18 years old and later claims to hold a supervisor position in her job. These checks are mostly based on self-reported information of the workers. In contrast, agreement meth-

ods or majority votes are outcome-based, meaning the quality of the results is solely assessed by analyzing the workers' submissions. While agreement mechanisms directly compare the submitted results and identify submissions with conflicts [47, 48], task workflows including majority votes redirect the output back to the crowd which votes for the best suited results [49]. Other works, e.g., the Find-Fix-Verify pattern introduced by Bernstein and colleagues [50], combines multiple of these mechanisms in one workflow.

To sum up, while several approaches exist to screen out poorly performing workers or research participants, such as using their reputation on the platform, instructional manipulation checks, task-specific gold standards, and consistency checks, these approaches have individual drawbacks. For example, some of these mechanisms are fakeable, some suffer from low reliability and validity, some are task-specific, and thus, need to be devised on a per-task basis, and some cannot be applied prior to the actual task. One approach, avoiding the need for testing workers prior to each task type and avoiding having to generate task-specific control workflows and gold standards, is assessing more general and context-independent skills or abilities such as intelligence or attention. So far, little research has been done on assessing the attention of crowd workers and/or research participants. Initial studies on the attentiveness of workers were performed by Hauser and Schwarz [51] and Goodman et al. [35]. By using instructional manipulation checks, they found that crowdsourced and non-crowdsourced participants did not differ in attention when following instructions. Nevertheless, even if workers read the instructions carefully and the tasks are easy, after a while, these tasks tend to become tedious due to their repetitive nature, resulting in low-quality work [52]. Rothwell et al. [53] showed that workers who were filtered in advance, for example, by reputation or gold standard tests, and workers who were not filtered in advance, do not clearly differ in attentiveness. Due to inattentiveness when reading instructions, workers in both groups equally failed the attention checks. Additionally, Peer et al. [54] highlight that attention checks may affect the outcome of tasks negatively.

One of the reasons for the scarcity of the research on the assessment of the workers' attention is that existing attention tests are not usable in a crowdsourcing environment. Therefore, the work focuses on the development and evaluation of an attention test for filtering workers a priori on their attention. The test considers diverse challenges arising due to its application in a crowdsourcing environment.

2.1.3 Aspects of Task-based Quality Improvements

As discussed in Section 2.1.1, considering the workers' needs while creating tasks is challenging. Especially due to the diverse facets which motivate the crowd to participate in tasks and the high dimensional process of designing tasks. The study of Schulze et al. [24] in 2011 gives first details about task properties that are important for workers with different demographics while selecting tasks. An overview of the evaluated task properties is given in Table 2.1.

According to [24], workers from the US are more interested in enjoyable, simple, and short tasks from well-reputed requesters. In contrast, workers from India are more focused on reward-related aspects, such as earning bonuses for good performance. Based on their findings, Schulze et al. [55] proposed a model of the task selection process and a concept for further investigation of the subject. However, their studies so far are limited to workers from one platform – MTurk. Hence, it is still not analyzed if the preferred task properties are comparable to task characteristics from other microtasking platforms and if the preferences change over time. Reasons for changes of the perception may be new features or types of tasks in the platform and the high fluctuation within the crowd, leading to newly, until now unidentified task properties. This work takes first steps to fill this gap, by collecting opinions from workers of different platforms and analyzing differences between the preferences of the workers as well as between the results and the findings of Schulze.

Table 2.1: Task properties investigated concerning their importance during the task selection process [24].

Category	Task Property
Task	Multiple tasks available Short time for task completion Task sounds interesting/enjoyable Simplicity of task Challenge of task
Payment	High reward per hour Bonus for good performance
Description	Examples of correct/incorrect answers Terms for rejection specified Background information about work Short task description Good language of description
Requester	High reputation of requester Ability to contact requester

Various research exists about how to setup and run tasks considering factors related to the performance of the crowd – a factor important for employers and workers. Going into detail of all these approaches would go beyond this work. An obvious factor which should be considered and reduced is the task complexity. A reduction of complexity can be achieved by providing clear instructions and simplifying the task through decomposition. Decomposition means to split large, complex tasks into short and easy to solve sub-tasks which are preferred by workers [39]. Even if the knowledge about the impact of the decomposition is little, several tools for an efficient decomposition are available, e.g., Curio introduced by Law et al. [56] and Turkomatic developed by Kulkarni et al. [57]. Both tools do not solely provide functionality for decomposing tasks, Turkomatic, for example, focuses also on the design of task workflows created by the workers themselves. Further, Jiang and colleagues [58] formally describe and analyze the

efficiency of the decomposition of tasks into dependent and independent sub-tasks while considering different cost schemes. However, their work provides no empirical study about the impact of task decomposition on the quality of work results. Instead, the outcome is limited to the impact of payment schemes, that differ from usual payment method in crowdsourcing platforms, in combination with task decomposition. Other research directions evaluated how to group already decomposed sub-tasks with respect to the reliability of results and cost efficiency [59]. Due to the limitation of previous studies, e.g., investigating the impact of the degree of decomposition using sub-tasks with nonuniform interface designs or employing the same test group for all sub-tasks which may lead to training effects, this work conducts an empirical study considering best practices while designing the sub-tasks to further investigate the impact of the decomposition on the quality of work results.

Best practices and guidelines for a wide range of different tasks, e.g., [60], highlight the necessity of including examples into the instructions, providing training tasks [61], and keeping instructions as simple as possible [62]. Other approaches to improve instructions and to reduce ambiguities directly involve the workers into the creation process of the task description [63, 64]. Both workflows iteratively identify cases of ambiguity and eliminate misunderstandings. However, poor results caused by misunderstandings due to language barriers can not be avoided. Providing the description of the expected work in native language of the workers may overcome this issue. Khanna et al. [65] show that this approach lowers the language barriers. In their study which is limited to inexperienced low-income worker from India, the improvement of the quality of working results is not solely caused by providing instructions in mother tongue. Instead, the combination of several approaches, e.g., including examples, leads to the observed increase of quality. As the quality may include different aspects depending on the type of task, an increase may be observed in different ways. This could be, for example, for tasks requesting to create text, an increase of the diversity of the results. Jiang et al. [66] found evidence about the existence of a relation between instructions and the diversity of task results for paraphrase

collection. However, these studies focus solely on the impact of the instructions on the quality of work results. It is still unclear, if there is also a relationship between the task processing language and the quality.

Therefore, in this chapter the impact of the language of instructions as well as the task processing language is evaluated. Besides lowering language barriers while reading the instructions, processing tasks in native language, e.g., tagging images or writing articles which are typical crowdsourcing tasks, may also significantly improve the quality of work results in terms of correctness and diversity.

2.2 Using Attention Testing for Quality Assurance

Existing quality assurance mechanisms suffer from individual drawbacks, outlined in Section 2.1.2. To overcome the disadvantage of their dependency on the task content, a mechanism which is based on non-fakeable characteristics of the workers might be a solution. Such characteristics could be, for example, the attention of the workers. Therefore, the applicability of a psychometric attention test as a selection mechanism is investigated. The newly developed test is applied prior to the actual task, it is fairly context independent and hence works for many ensuing crowdworking tasks, and it cannot be faked up. Besides the applicability, the predictive potential of the test and the filter validity are evaluated. Figure 2.2 gives an overview about the conducted studies for the test evaluation.

2.2.1 Introduction to Attention Testing

Attention is a basic cognitive function. It is defined as the sustained focus of cognitive resources on relevant stimuli while ignoring irrelevant stimuli. Although psychology distinguishes different forms of attention, this work is concerned with sustained selective attention, also called concentration. Sustained selective attention is the ability to maintain a consistent behavioral response during a

Section	Evaluation	Test version	Study participants
2.2.3	Applicability and reliability	1	1313 participants from WiSoPanel, Microworkers and Figure Eight
2.2.4	Predictive validity	1 & 2	565 testtakers participating in study on applicability (Section 2.2.3)
2.2.5	Filter validity	2	144 participants from Microworkers

Figure 2.2: Overview about test evaluation.

continuous and repetitive activity. During this activity, a person tries to detect the appearance of a target stimulus while suppressing a response to non-target stimuli. Crowdsourcing tasks usually require a certain level of concentration on the worker's side, as the tasks are often simple but repetitive. That is why attention tests lend themselves to the context of microtasking. There are many attention tests such as KLT-R [67], CAPT [68], IMT/DMT [69], and d2-R [70]. These tests differ in the complexity of their setup and in the degree to which they are established and validated. Being attention tests, however, they share the characteristic that the basic task, e.g., identifying target stimuli, is easy. Their independence of intelligence makes the tests applicable to a large segment of the population. A low attention test score indicates a low degree of attentiveness, either resulting from a low ability to mentally focus or low motivation or ability to take the test, e.g., not understanding instructions or motor impairments. The idea is to make use of an attention test in a crowdsourcing setting to distinguish workers who are attentive from those who are inattentive. For the practical purpose of identifying good versus bad workers, it is not necessary to figure out why an individual worker scored low on the attention test. There are requirements of any attention test that is to be used in crowdsourcing or to be used in an online setting more generally: Workers are paid per time, so the test must be short, including its warm-up phase. Moreover, due to the global distribution of

crowdsourcing workers [71], the instructions must be provided online without personal interaction and must be understandable to users of various languages and diverse cultural and educational backgrounds. Consequently, the test should not require a superb command of the language the test is in, instructions should be easy to follow, and the test itself should tap pure attention without requiring further skills on the part of the worker, e.g., calculus. Furthermore, the test should not require the worker to install any software, e.g., browser plug-in, and should not require more than average hardware. The latter two requirements imply that text-based attention tests are particularly suitable. Lastly, the test should feature a large and broad norm sample: When testing only one sample from any given platform, the sample's attention score must be related to a suitable norm sample to get a sense of the level of attentiveness on this platform. When perusing available offline tests, e.g., KLT-R, CAPT, IMT/DMT, and d2-R, to see whether they meet the requirements of an online attention test, most tests do not fit the purpose of assessing attention in an online environment. For example, KLT-R takes too long, CAPT has a narrow norm sample and is not text based, IMT/DMT takes too long and captures impulsiveness in particular. The d2-R test comes close to fulfilling the requirements. The test is a revised version of the original d2 test by Brickenkamp [72]. It is a validated and well-established test of sustained selective attention. It is a paper-and-pencil cancellation task in which participants find and cross out any letter d with two marks surrounding it in a field of 14 rows, each row with 57 letters. The time for processing a row is 20 s. The distractors are similar to the target stimulus; for example, a "p" with two marks or a "d" with one or three marks. While it is not language free, the test does not require a superior command of the test language, does not require elaborate cognitive skills, is text-based, is short, and a norm sample is available. Hence, the newly introduced test was inspired by the method of the d2-R [70].

Running an attention test online implies that it is administered in an unsupervised manner. Due to the absence of an instructor, the test should include detailed yet clear instructions. As the developed attention test is web-based, the worker or a proband without crowdsourcing background, further referred

as *testtaker*, does not need to install any software or needs more than average hardware. Furthermore, testtakers may vary in the device on which they take the test. Differences in screen size or input mode may influence testtakers' performance. Moreover, the particular surroundings and the situation in which testtakers complete the test may lead to distraction as well as hidden influences such as workers' disabilities might affect test results. In addition, crowdsourcing-specific challenges must be considered. The rewards earned through completing the task may encourage tricking the system or rushing.

2.2.2 Description of Attention Test attentiveWeb

Based on the method of d2-R, the online attention test attentiveWeb is developed; however its realization differs from d2-R. In attentiveWeb, the letters are realized as buttons labeled with the targets and distractors. The buttons are arranged in rows separated by small spaces between the buttons. By clicking the buttons, the letters are marked as being crossed out. A button click cannot be reversed. The size of the display area is fixed, so the row of buttons is not wrapped on small displays. In addition, the size of each testtaker's display is assessed. If the display happens to be too small, the participant is prompted to use a larger display to accommodate an entire row of buttons.

Two versions of attentiveWeb are created as follows: In Version 1, the target letter is *d* and the distractor letter is *p*, whereas in Version 2, the target letter is "b" and the distractor letter is "q". In contrast to d2-R, not all rows are visible at once, but one row per screen is shown at a time. This allows enforcing a maximum working time of 20 s per row. After 20 s, a pop-up notifies the testtaker, and input is disabled. If the testtaker needs less than 20 s to process a row, a submit button enables manual submission. After a countdown of 3 s, the next row is displayed automatically. In total, the testtaker completes 14 pages with one row each. As feedback on the progress, the number of processed and number of total rows are shown at the bottom of each page. Figure 2.3 depicts a sample of

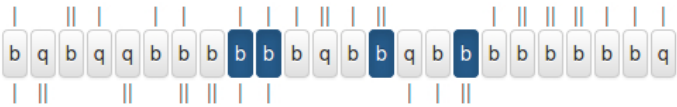


Figure 2.3: Sample of buttons shown on each test page of attentiveWeb.

buttons shown on each test page. Instructions are presented at the beginning of the test. The instructions are given in simple English.

The custom metrics for evaluating a testtaker's performance in the benchmark d2-R are the concentration performance (CP) and the error percentage (E%). These metrics rest on the number of clicked target items TN and the number of errors. The errors are either omission errors $E1$ or confusion errors $E2$.

$$CP = TN - (E1 + E2) \tag{2.1}$$

CP considers the number of processed target items TN and both types of errors, see Equation (2.1); thus, it represents the quantity of correct performance given the available time. For the calculation in attentiveWeb, just as with the d2-R [70], the first and the last row are omitted. Thus, the maximum of CP is approximately 310 with a ratio of target and non-target items of 5:6. As the total number of errors $(E1 + E2)$ can be larger than the number of clicked target items TN, CP may be negative.

$$E\% = \frac{E1 + E2}{TN} \cdot 100 \tag{2.2}$$

E% is the sum of errors $(E1+E2)$ divided by the number of clicked target items, see Equation (2.2); thus, it represents inaccuracy. As the total number of errors can be larger than the number of clicked target items, E% ranges from 0 % to more than 100 %. Again, the first and last row are omitted for the calculation.

The metrics in d2-R require that the testtaker has worked on the test, which means that at least one target item should have been processed per row. Ad-

ditionally, the computation of the d2-R metrics assumes that the testtaker processes the items from left to right. Due to the unsupervised administration of the test, the evaluation metrics in attentiveWeb were made more robust. As there is no guarantee that each participant works on each row beginning on the left, the start point in each row is determined. Further, CP and E% were adapted to tolerate incomplete page submissions as follows: In each row, it is distinguished if the testtaker has processed no item at all or if s/he has clicked at least one button. If no button has been clicked, TN is calculated by the total number of target items shown in the row; hence, the number of omission errors equals TN. If no item has been processed, the number of confusion errors is zero. Then, CP is zero, and E% is 100 %. If at least one item has been processed, the metrics must be adapted only if no target item has been clicked. If the testtaker failed to click any target item, TN is the number of target items from the first button to the last-clicked non-target item, and the number of omission errors is the number of unprocessed target items until the last-clicked non-target item. An adaptation of $E2$ is not necessary. When using the adapted values of TN and $E1$, CP equals $E2$, and E% is larger than 100 %.

As a minimal filtering criterion when using attentiveWeb for quality control, CP larger than zero and E% below 100 % should be used. Failing this check means, that less target items than non-target items were clicked, indicating that the participant did not understand or boycott the test. It is also useful to exclude workers who did not process each row, meaning they click at least one button per row. By adapting the accepted minimal concentration performance and error rate the filter mechanism can be individually adapted to the needed attentiveness depending on the task characteristics and requirements.

2.2.3 User Studies on Applicability

To (1) illustrate the applicability of the developed attentiveWeb, (2) determine its reliability, and (3) evaluate the attention of workers in two microtasking plat-

forms and in one online panel, three parallel user studies were conducted. In this section, first the studies are described and then the results are presented.

Studies' Description

To collect additional information about the testtakers, two questionnaires were added to attentiveWeb (Version 1 with target letter "d") – one given prior to and one given after the attention test – to capture potential influences on the attention. In the first questionnaire, the testtakers indicated their gender, age, country of residence, and where they were completing the test. Moreover, the participants self-report their state of attentiveness and mouse skills, as both might correlate with the attention test results. In the second questionnaire, the testtakers indicated their state of attentiveness again and provide information if s/he did the d2 test before. The full set of questions is in Appendix B.1.

The user studies were fielded from May 30 to June 3, 2016. The participants were recruited from three platforms: (1) Microworkers, (2) Figure Eight, and (3) WiSoPanel, an online panel that holds Germans from all walks of life who have agreed to take part in noncommercial web-based studies [73]. On the two crowdsourcing platforms, workers were solicited by paid tasks. To prevent that all available tasks were taken within the first hours, the tasks were released successively. Otherwise, the tasks would have been available only to users from specific time zones. The task was announced as taking 10 min to 15 min with a reward of US$0.20. This corresponds to a typical payment on crowdsourcing platforms [74]. In total, 420 participants from Microworkers submitted their answers to the final questionnaire. On Figure Eight, 308 workers finished the test. In WiSoPanel, too, the task was announced to take 10 min to 15 min, but with a reward of EUR 0.50. The participants from the WiSoPanel are used to a higher payment than is customary in crowdsourcing platforms, i.e., about EUR 3-4 per hour. Hence, compared to the two crowdsourcing platforms, a customary but unequal reward over an equal but unusually low reward was favored. To mimic the wave-like manner in which the crowdsourcing workers were solicited, 12 237

WiSoPanel users were invited via e-mail in four waves. Of those, 1 352 submitted their answers to the final questionnaire.

For the evaluation only testtakers with $CP > 0$, $E\% < 100\,\%$, and at least one clicked button per row are considered, resulting in 940 users from WiSoPanel, 183 workers from Microworkers and 190 participants from Figure Eight. When comparing the three platforms, the smallest share of participants is filtered out in Figure Eight (57 % of workers are left), the second smallest share is filtered out in WiSoPanel (51 % are left), and the highest share is filtered out in Microworkers (34 % are left). With Pearson's chi-squared test the relationship between the extent of filtering and the platform is examined. The results show a significant association between these two variables ($\chi^2(2) = 93.063$, $p < 0.001$), meaning that the number of users passing the filter on Figure Eight and WiSoPanel is significant larger than on Microworkers.

The analysis of the answers of the first questionnaire shows that WiSoPanel holds primarily Germans (94 %). The two crowdsourcing platforms are more heterogeneous with regard to workers' country of residence. The three most frequent countries of Microworkers are Bangladesh, India, and Serbia. On Figure Eight, Serbia is most represented, followed by Venezuela and India. Comparing the two crowdsourcing platforms, the users from Microworkers originate mainly in Asia, whereas the workers from Figure Eight are more international.

Figure 2.4 presents the distribution of age-groups, as far as specified by each testtaker. Due to a larger number of participants from WiSoPanel who are older than 50 years, these testtakers are grouped in distinct groups. While the participants from the crowdsourcing platforms older than 50 years are grouped all together. With regard to age in WiSoPanel, age varies more widely, and the median age-group, i.e., 41–50, is higher than in the two crowdsourcing platforms (Microworkers: 20–30 and Figure Eight: 31–40). By using the Kruskal-Wallis test, it is examined if the three samples originate from the same age distribution. The test results show that the underlying distributions differ ($\chi^2(2) = 389.06$, $p < 0.001$).

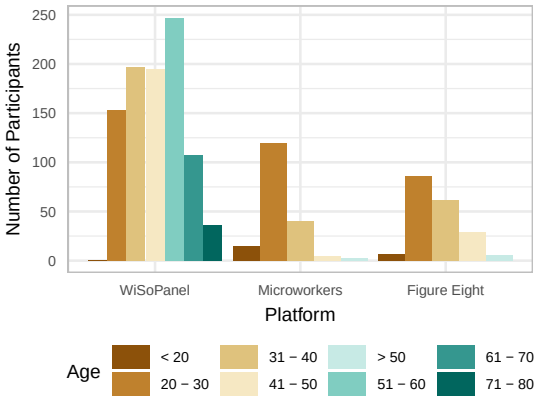


Figure 2.4: Age-group by platform.

With regard to gender, a slight majority of users recruited from WiSoPanel are women (55 %). In contrast, most workers on the two crowdsourcing platforms are men (Microworkers: 77 % and Figure Eight: 69 %). The inequality of the gender distribution across the three platforms is significant ($\chi^2(2) = 91.35$, $p < 0.001$).

As it was known that most of the WiSoPanel users live in German-speaking countries [73], the attention test was presented in German. Due to the international audience, the working language of most crowdsourcing platforms is English. That is why in the two crowdsourcing platforms, the attention test was in English. Given the reported countries of residence of the crowdsourcing workers, this implies that the attention test was not in most workers' mother tongue. The language barrier may lead to misunderstandings, and hence, the results obtained from the two crowdsourcing platforms may be affected by workers having worked in a foreign language, cf. Section 2.1.1.

Table 2.2 shows the participants' self-rated skills in the language of the test. As expected, most of the WiSoPanel users are German native speakers; thus, they

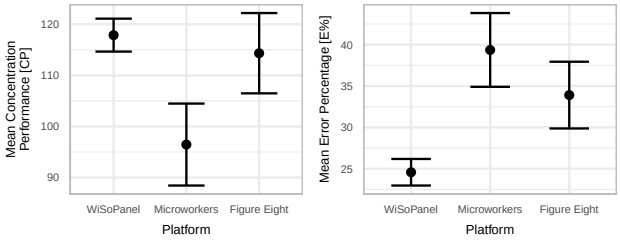
Table 2.2: Language skills by platform.

Language skills	WiSoPanel	Microworkers	Figure Eight
Beginner	1 (<1 %)	42 (23 %)	36 (19 %)
Advanced	30 (3 %)	117 (64 %)	135 (71 %)
Native speaker	900 (96 %)	24 (13 %)	18 (10 %)
Not specified	9 (<1 %)	-	1 (<1 %)

took the test in their mother tongue. On the crowdsourcing platforms, around one fifth of the workers self-identified as having poor English skills, and about two thirds reported having good English skills. By using Pearson's chi-squared test the distributions of the language from the two crowdsourcing platforms are compared, resulting in no significant differences ($p = 0.285$).

Evaluation of Attention

The attentiveness of the testtakers of the platforms is analyzed and compared by evaluating the achieved concentration performance CP and the error rate E%. Figure 2.5 presents the attention values by platform including 95 % confidence intervals. Figure 2.5a gives the mean CP. Users of the WiSoPanel obtained a mean CP of 117.9, those from Figure Eight 114.3 and those from Microworkers 96.5. For further analyses of the means of CP, first the variance is examined. By using Levene's test, an inhomogeneity of the variance across platforms is observed (Levene$(2, 1\,310) = 4.32$, $p < 0.05$). The source of the difference is between WiSoPanel and Microworkers (Levene$(1, 1\,121) = 7.41$, $p < 0.01$). Workers from WiSoPanel and Figure Eight perform more uniformly with regard to CP. Differences concerning the mean CP are evaluated with Welch's ANOVA which does not assume equal variances. The test results in significant differences among the three platforms (Welch$(2, 320.33) = 11.92$, $p < 0.001$). Tamhane's post hoc test for data with unequal variance revealed that Microworkers had a significantly lower average CP than WiSoPanel users (Tamhane $= 21.42$, $p < 0.001$) and Figure Eight workers (Tamhane $= 17.88$, $p < 0.01$), while WiSoPanel and Figure Eight did not differ ($p = 0.80$).



(a) Mean CP by platform. *(b) Mean E% by platform.*

Figure 2.5: Attention performance by platform with 95 % confidence interval.

Figure 2.5b shows the mean E% broken down by platform. The users of WiSoPanel worked most accurately with an average E% of 24.6 %. E% with Microworkers is 39.4 % and with Figure Eight 33.9 %. Again by using Welch's ANOVA, the means of E% are compared. Among the three platforms, E% differs significantly (Welch$(2, 311.88) = 24.93, p < 0.001$). Post hoc tests revealed that workers in WiSoPanel had a significantly lower E% than those in Microworkers (Tamhane $= 14.79,\ p < 0.001$) and in Figure Eight (Tamhane $= 9.83,\ p < 0.001$), while the crowdsourcing platforms do not significantly differ ($p = 0.21$). On the one hand, there are differences in the demographics of the workers of the three platforms, while on the other hand, there are differences in the attention test results among the platforms. To bring the differences in the demographics and in the attention test results together, stepwise regression analyses are conducted. In a first step, it is tested whether any known differences among the workers of the three platforms are relevant to their attention test performance using backward elimination of non-significant predictors. In a second step, it is examined whether accounting for attention-relevant differences among the workers of the three platforms sufficiently explains their attention test results or whether despite taking those differences into account there remain inherent platform-related differences with regard to attention. In Step 1 of each of the two regression analyses, all known predic-

tors were evaluated, namely, gender (women vs. men), age (up to 40 years vs. over 40 years), country (Western European culture [including European countries and the United States] vs. non-Western-European culture [all others]), language (native language vs. foreign language), current place (at home vs. elsewhere), people around (alone vs. not alone), self-rated mental focus before attention test (five levels), self-rated mental focus after attention test (five levels), self-rated mouse skills (five levels), and test done before (no vs. yes). CP was higher among younger testtakers, indicated by the standardized regression coefficient $\beta = -0.24$, among people who live in a Western European culture ($\beta = 0.26$), among those who participated not from home ($\beta = 0.06$) and with high self-rated mouse skills ($\beta = 0.08$). The finding of cultural differences ties in with Litman et. al [75] who observed that India-based workers submitted data of a lesser quality than US workers. Leaving these four significant predictors in the model and testing whether dummy-coded platforms explain additional variance in attention reveals that Figure Eight achieves a higher CP than the other two platforms together ($\beta = 0.12$), while WiSoPanel by tendency is higher than the other two platforms together, although this tendency fails a conventional level of significance ($\beta = 0.09$). The final model, which includes the four predictors from Step 1 and the dummy-coded platform variables, fits well, $F(6, 1\,287) = 20.49$, $p < 0.001$, adjusted $R^2 = 0.083$.

The error percentage E% was higher among older testtakers with a standardized regression coefficient $\beta = 0.15$ and among those who live in a non-Western-European culture ($\beta = -0.29$). Leaving these two significant predictors in the model and testing whether dummy-coded platforms explain additional variance reveals that WiSoPanel has a lower E% than the other two platforms together ($\beta = -0.14$), while Figure Eight is not lower than the other two platforms together ($\beta = 0.06$). The final model, which includes the two predictors from Step 1 and the dummy-coded platform variables, fits well, $F(4, 1\,287) = 23.94$, $p < 0.001$, adjusted $R^2 = 0.067$.

Using either attentiveWeb metric shows that workers' age impacted attention test performance and that performance profited from younger age. This likely

reflects the often made finding that performance in timed tests lessens with age because of declining perceptual motor skills [76, 77]. One can assume that CP profited somewhat more from younger age than the error percentage because age shifts the speed-accuracy trade-off toward accuracy [78]. Moreover, users who participated from elsewhere but their home had a better performance because of self-selection pertaining to motivation: Those who decide to comply with a work request despite having to deal with the discomfort of not being at home have a stronger motivation to participate and consequently attain better results. The better performance with higher self-evaluated mouse skills is self-explanatory. There were no influences of gender on either of the two metrics; hence, the different gender proportions across the three platforms had no bearing on observed differences in attention. Moreover, the varying levels of language skills had no effect on the attention test performance, since any possible influence of language skill likely had been absorbed by the variable coding for culture. Finally, it did not matter if a worker had previously done such kind of attention test.

To determine attentiveWeb's reliability, first the internal consistency is examined with Cronbach's α. Second, the split-half reliability is calculated to further analyze the homogeneity of the test. For the calculation of Cronbach's α each test was split into four parts containing three rows. Each of the resulting quarter of attentiveWeb was treated as an item. In WiSoPanel ($n = 940$), attentiveWeb's internal consistency was 0.91 with CP and 0.87 with E%. In Microworkers ($n = 183$), Cronbach's α was 0.89 with CP and 0.85 with E%. In Figure Eight ($n = 190$), Cronbach's α was 0.90 with CP and 0.87 with E%. The internal consistency with CP did not differ among the three platforms, $\chi^2(2) = 1.75, p = 0.42$, nor did the internal consistency differ among the platforms with E%, $\chi^2(2) = 1.78, p = 0.41$ [79]. Thus, across all platforms together ($N = 1\,313$), Cronbach's α was 0.91 with CP and 0.87 with E%.

Regarding split-half reliability, in WiSoPanel ($n = 940$), it was 0.90 with CP and 0.85 with E%. In Microworkers ($n = 183$), split-half reliability was 0.85 with CP and 0.85 with E%. In Figure Eight ($n = 190$), split-half reliability was 0.87

with CP and 0.85 with E%. The split-half reliability with CP did not differ among the three platforms ($\chi^2(2) = 2.68$, $p = 0.26$), nor did the split-half reliability differ among the platforms with E% ($\chi^2(2) = 0.01$, $p = 0.99$). Thus, across all platforms together ($N = 1\,313$), split-half reliability was 0.89 with CP and 0.86 with E%. Overall, the analysis establishes the internal consistency and split-half reliability of attentiveWeb.

To sum up the comparison of the platforms, despite accounting for attention-relevant differences among workers on the three platforms, there remained inherent differences across platforms with regard to attention. These are marked demographic differences of workers in an online panel and crowdsourcing workers, and marked differences in the quality of their work results, tie in with Smith et al. [80] who compared workers in an online panel with MTurk. An explanation for the observed work quality differences might be the types of tasks that are typically offered on these platforms. Perhaps users of Figure Eight and WiSoPanel are more used to tasks that resemble the attention test than are Microworkers. Furthermore, online panels and crowdsourcing platforms differ in characteristics that might be relevant for their users' motivation or ability when carrying out a task [81]. In online panels, people have expressed their interest in participating in web-delivered research studies, whereas in crowdsourcing platforms, people have expressed their interest in carrying out different kinds of web-delivered work more generally. The motivation to carry out special work for which one has expressed interest is likely to be higher than the motivation to carry out work that falls within a broad spectrum of work for which one has signed up. Moreover, participating in research studies, unlike in various kinds of web-delivered work, promises to fulfill motives such as being entertained, learning something about oneself, or contributing to discover scientific facts. Seeking to have such motives fulfilled by one's participation makes submitting sloppy work or cheating quite pointless. By contrast, workers of a large crowdsourcing platform described the platform as a labor market [82], and Litman and colleagues [75] showed that the motivation of crowdsourcing workers has shifted from being primarily intrinsic to being mainly extrinsic. Furthermore,

as most online panels are smaller than crowdsourcing platforms, they represent close-knit networks, and their participants are likely to be treated more personally. Thus, participants in online panels may have a stronger identification with and sense of membership in the platform than participants in crowdsourcing platforms.

2.2.4 User Studies on Predictive Validity

To establish the predictive validity of attentiveWeb, follow-up studies were conducted more than two years later with the same testtakers participating in the first analysis of attentiveWeb, cf. Section 2.2.3.

Study Description

For the validation studies, Version 2 of attentiveWeb is used, wherein all d's are replaced by b's and all p's by q's. Everything else was kept the same as in the previous user studies except for removing a few items in the two surrounding questionnaires whose answers were unlikely to have changed in the meantime such as gender and age, cf. Appendix B.1. The validation studies were fielded September 4-16, 2018. Participants were recruited by inviting those participants on Microworkers and on WiSoPanel who had taken attentiveWeb (Version 1) as part of the user studies more than two years earlier. It was impossible to resolicit the workers from Figure Eight, because the platform had changed its cost scheme. Overall, 25 of the 183 eligible Microworkers submitted the final questionnaire. Of the 940 eligible people from WiSoPanel, 540 finished the validation study.

Evaluation of Predictive Validity

Applying the filtering based on the concentration performance CP and error rate E%, 21 (84.0 %) of the participants from Microworkers passed the check. From WiSoPanel 453 (83.9 %) invited testtakers passed the attention filter.

Based on the results of these participants, in WiSoPanel, attentiveWeb's predictive validity was $r = 0.58$ with regard to CP and $r = 0.40$ with regard to E%. In Microworkers, the validity was $r = 0.48$ with CP and $r = 0.38$ with E%. To analyze if the difference between the correlation coefficients is significant, the values are transformed to their z-scores with Fisher's r to z transformation. The significance is examined on a level of 0.05. The validity with CP did not significantly differ between the two platforms, $z = 0.58$, nor did the validity with E%, $z = 0.09$. In the two platforms together, attentiveWeb (Version 1) predicted crowdworkers' performance more than two years later in attentiveWeb (Version 2) at $r = 0.57$ with CP and at $r = 0.40$ with E%.

The attentiveWeb validity test at hand used a much longer retest interval of more than two years, and the validity test at hand was not based on re-administering the same task but on administering a similar attention task. Thus, the validation studies tested for predictive validity rather than retest validity. Predictive validity is lower than retest reliability because predictive validity refers to a different test taken at a later time and retest reliability refers to the same test taken at a later time. Additionally, reliability tends to be higher the shorter the retest interval, except for very short retest intervals where fatigue may play a role. Hence, attentiveWeb's predictive validity of 0.57 with CP is to be considered very good and the 0.40 with E% outstanding. Applying attentiveWeb on workers allows for predicting their attention more than two years later.

The fact that attentiveWeb's internal consistency and split-half reliability have been established on three platforms (WiSoPanel, Microworkers, and Figure Eight) and its predictive validity on two platforms (WiSoPanel and Microworkers) lends confidence in its quality when used on other crowdsourcing platforms as a personnel selection test for tasks that require sustained attention. Furthermore, attentiveWeb has proven its robustness: It showed similar test quality when used on several platforms, wherein it was administered in different language versions, i.e., English and German, and to audiences that differed in demographics and in other characteristics.

2.2.5 User Study on Filter Validity

In the last step of the evaluation of the applicability of attentiveWeb for quality control, the filter validity is analyzed. To do so, it is evaluated by a pilot study if workers passing the attentiveWeb test submit high quality results. The design of the pilot study is described first, and then the results are presented.

Study Description

The validity of the filtering is investigated for two types of tasks, combined in one task. The first part of the task is a typical job on microtasking platforms – a short transcription task, where the workers have to digitalize two short, handwritten texts[1]. During the transcription music is playing in the background. The playback is interrupted after a while. After finishing the transcription, the workers are asked if they noticed the interruptions, resulting in the second task type – a subjective study. This type of task is selected, as crowdsourcing is often used to recruit participants for Quality of Experience (QoE) studies, for example, to evaluate the quality of video streams [60]. Combining the transcription task and the subjective study allows to compare the effect of the filtering for different kind of tasks for the same test group.

The pilot study was fielded on Microworkers in July and August 2019. Participants were rewarded with US$0.30 and the expected length of the task was less than 7 min. Those who submitted at least one out of the two texts were invited to the attentiveWeb test with a time shift of a few hours. This shift has no impact on the results of attentiveWeb, due to its established predictive validity.

Overall, 185 workers submitted at least one digitalized text. Of those 144 completed the attentiveWeb test. The participants are mainly male (72.9 %) in the age between 20 and 30 years (50.4 %). They are from all over the world (43 countries) with India 27.1 % as the most occurring country of residence.

[1] The texts were taken from IAM Handwriting Database http://www.fki.inf.unibe.ch/databases/iam-handwriting-database; Accessed: August 1st, 2020

Table 2.3: Correlation coefficients for the quality of work results, i.e., transcription quality and recognition of music stops, and the results of attentiveWeb.

Task	CP	E%
Transcription text 1	0.283^{***}	-0.251^{**}
Transcription text 2	0.259^{**}	-0.238^{**}
Recognition of music stops	0.211^{*}	-0.264^{**}

$^{*}\ p < 0.05$, $^{**}\ p < 0.01$, $^{***}\ p < 0.001$

Evaluation of Filter Validity

For the evaluation of the filter validity, the correlation between the results of each task type and the measured attentiveness of the participants is analyzed. To quantify the quality of the transcriptions from the first part of the task, Levenshtein's editing distance with normalization is used. Levenshtein's distance describes the number of insert, delete and transform operations needed to convert one string into another. The distance is normalized by dividing the distance by the length of the longer string. For the analysis of the study on recognizing interruptions in audio streams and the attentiveness the point-biserial correlation is computed. In the evaluation, the results of 34 participants, who reported that they did not listen to the music, were excluded. The remaining workers are grouped based on their answer about the recognition of the interruptions of the music.

For both tasks the quality of work results (quality of work results) and the attentiveness of the workers are correlated. The correlation coefficients with level of significance for both attention metrics – CP and E% – are presented in Table 2.3. The magnitude of Pearson's correlation is low to mid for Levenshtein's distance of the transcribed texts and the concentration performance as well as the error rate. This means, workers submitting results with higher quality achieve a higher performance in concentration. Further, they make less mistakes while processing the test attentiveWeb. Even if CP and E% are highly negative correlated (Pearson: –0.928), the concentration performance and the transcription quality are slightly stronger correlated than the error rates and

the quality. Furthermore, the correlations related to the first text are slightly stronger than the ones from the second text.

The correlation between the awareness of the music stops and the values representing the workers' attention is also significant. Thus, study participants who noticed the music interruptions have a higher concentration performance and made less mistakes than workers who did answer that the music did not stop. Other than by the transcription task, the quality and E% are stronger correlated than the quality and CP. Reason for this observation may be that workers who work more carefully on tasks leading to less mistakes – a lower error rate – are more aware of short-lived events than workers who are focused on the working speed.

Thus, the filter validity of attentiveWeb is established. Filtering workers based on their attentiveness lead to task results with higher quality than using no control mechanism. One can make another observation, the filter also excludes workers who provide results of good quality but do not achieve a high concentration performance in the attention test. This effect may be caused by the independence from task characteristics and the work subject. However, the share of excluded workers, and thus, the strictness of the control mechanism, can be managed by the required filter criteria, i.e., the accepted concentration performance and error rate.

2.3 Task Selection - The Workers' Perspective

Designing tasks while considering the worker's perspective is by far more complex than from the employer's point of view. Here, not only the extrinsic motivation plays a significant role, also the intrinsic motivation influences especially the task selection of the workers. The task selection in turn influences the overall completion time of tasks as well as the quality of the work result. Schulze et al. [24] investigated relevant task properties for task selection from the perspective of users from MTurk in 2011. Besides appropriate payment and allocated time, clear instructions are the most important task characteristics for

the workers. The ranking of these properties depends on demographics of the workers. Yet, it is still unclear if their preferences are robust over time. Fluctuations of active members of microtasking platforms and resulting changes in user characteristics, demographics and behavior, may also cause changes concerning task preferences. Further, previous studies have only focused on users from one platform. The perceptions and preferences may differ between platforms.

This section sheds light on the preferences of the workers concerning various task properties from two large platforms – MTurk and Microworkers. The data is collected by an empirical study which is partly based on the study of Schulze et al. [24]. Besides the analysis of the users' preferences, the findings are evaluated concerning similarities and differences between the two platforms.

2.3.1 Survey Description

The goal of the survey is to analyze the relevance of different task characteristics, e.g., instructional aspects, and to identify relevant design aspects with respect to their impact on workers while selecting and working on tasks. Further, design preferences of the users of both platforms MTurk and Microworkers are analyzed and compared. The survey consists of three major parts, which are described in detail in the following. First, the workers are requested to provide some demographic data, e.g., basic aspects like gender, age, and level of education. Apart from this, they are asked for qualification-related information, i.e., their approval rate, as well as for more general matters, e.g., how long they have already been working on the platform and which kind of tasks they are most interested in.

The second part of the survey concerns general aspects of task design. This part of the study is based on the empirical study by Schulze et al. [24] about relevant task characteristics. In addition to the nine properties identified by Schulze et al., e.g., appropriate size of salary, clear instructions, high reputation of requester, and the time allocation, two further task characteristics are incorporated into the survey: the graphical design of the task and the ease of use of

the page design, see Appendix B.2. The latter corresponds to the usability of the task but is circumscribed in order to avoid confronting the workers with any technical terms they may not understand. The resulting task properties are then transformed into contrasting aspects by formulating each one in a positive way in the one case, and in a negative way in the other case. For the positive formulations, the workers are asked to rate the extent to which they consider the given aspects to be important or unimportant for task selection on a 5-point Likert scale. Respectively, they are requested to rate the extent to which they regard the negative formulated properties as frustrating (or not) on a 5-point Likert scale.

The third section focuses on a set of quite specific aspects of task design that can mostly be reduced to concrete design decisions. For this purpose, the workers are presented 14 different starts of records like "I usually prefer…", each of which is followed by two alternate endings of the sentence. For each pair of alternatives, they are instructed to choose the one that intuitively best fits their preferences. Some of the aspects taken up at this point touch findings of previous, crowdsourcing-related literature. One, for example, addresses the question whether embedded Q&A would be a helpful feature from the workers' perspective. It is inspired by the work of Brewer et al. [83] who proposed that embedding Q&A could provide real-time feedback. The main objective of this section is to gain a general overview of the workers' preferences with respect to selected, high-level design decisions as well as to gather some further information regarding their task selection habits.

2.3.2 Survey Conduction

The survey was conducted on two large, international platforms MTurk in April 2018 and Microworkers in May 2019. It was embedded into both platforms using their internal template functionality. As some of the qualification-related questions are platform-specific, the survey conducted on MTurk contains more questions and was designed to be completed within 20 min. Thus, for their par-

Table 2.4: Overview of the workers' demographics.

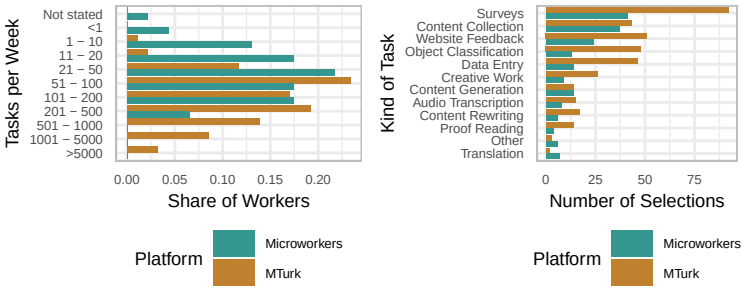
Category	Value	# Workers MTurk	# Workers Microworkers
Gender	Female	33 (35 %)	18 (39 %)
	Male	61 (65 %)	28 (61 %)
Age	< 21	-	2 (4 %)
	21–30	40 (43 %)	15 (33 %)
	31–40	36 (38 %)	14 (30 %)
	41–50	5 (5 %)	9 (20 %)
	51–60	10 (11 %)	6 (13 %)
	61–70	3 (3 %)	-

ticipation the workers on MTurk were paid US$3.50, and on Microworkers the reward was US$3.00 due to the slightly shorter questionnaire. To reduce the risk of misunderstandings due to language barriers, the task access was restricted to US only. In total, 105 workers from MTurk and 53 users from Microworkers participated in the survey.

To ensure the quality and validity of the subjective responses, two measures were taken. First, all workers who missed to fill in two or more of the obligatory elements, were excluded. While it may happen that one overlooks a form element by mistake, two or more absent answers may be an indicator of inattention. Overall, ten workers from MTurk were affected by this measure.

Second, the answers given to the questions about the number of completed tasks and income were evaluated. While the provided information of the workers from Microworkers were validated by comparing the values to the working statistics displayed on their public worker profiles, the average reward per task of participants from MTurk had to be reviewed manually. Due to mismatching values seven participants from Microworkers were excluded. Only for one worker from MTurk the average reward was beyond any conceivable size. Further, this participant provided solely nonsensical answers to the free-text questions. Ultimately, 94 workers from MTurk and 46 participants from Microworkers remained for the evaluation.

Table 2.4 shows an overview of the distribution of the workers' demographics which is mostly in line with the distributions shown by Martin et al. [71].



(a) Number of tasks per week.

(b) Preferred kind of tasks.

Figure 2.6: Information about working statistics and preferences provided by the participants.

Regarding the workers' level of experience with the platforms, 76 % of the participants from MTurk have already been working for one year or longer while only 46 % of the workers from Microworkers answered to be that long active. While only 3 % of the workers from MTurk reported to use the system for less than a month, 25 % of the users from Microworkers just started working on the platform. Overall, this indicates that the participants from MTurk are a bit more familiar with the platform.

This finding is also supported by the average number of tasks that the workers completed per week when considering the last month, which is depicted in Figure 2.6a. The figure shows the average number of weekly completed tasks against the share of workers. The answers of the two platforms are presented in different colors. While the shapes of the distributions are similar, there is a shift towards a higher number of completed tasks for workers from MTurk. This is also indicated by the differences in the second quantile. The median number of tasks reported by the workers is higher for participants from MTurk (101–200 tasks) than for workers from Microworkers (21–50 tasks). Finally, the workers were asked to select all jobs they usually work on out of a list of task cate-

gories. The results are illustrated in Figure 2.6b. As one might expect, surveys are the most selected ones on both platforms. The categories chosen second, third, and fourth most frequently from MTurk users lie close together and comprise website feedback (51), object classification (48), data entry (46), and content collection (43). Workers from Microworkers selected the categories content collection (37) and website feedback (24) second and third most frequently. Apart from the categories listed above, all others were only selected by about a third or less of the participants from Microworkers and a quarter or less of the workers from MTurk.

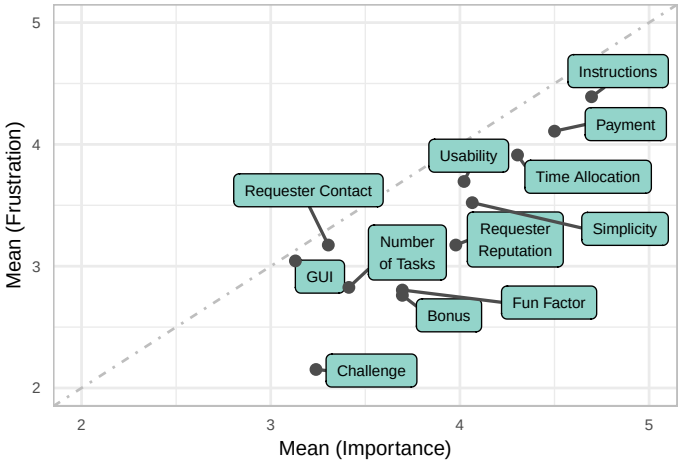
2.3.3 Evaluation of Survey Results

In this section the importance of different task properties based on the answers given to the surveys is evaluated and discussed.

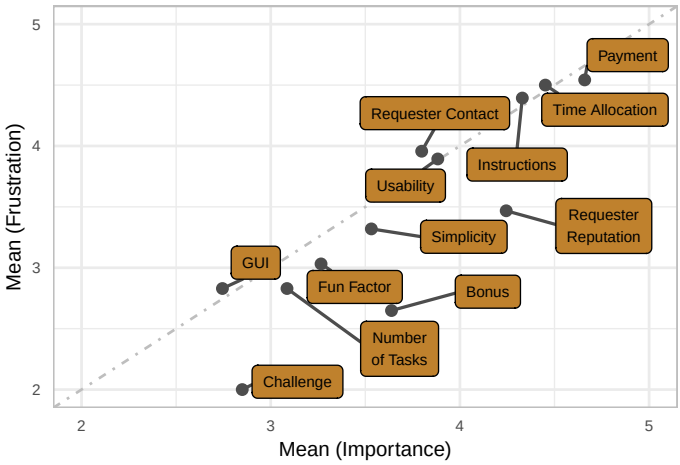
Relevance of Task Properties

The goal of this section is to gain insight into the relation between the importance of certain task properties for task selection and the extent to which a poor implementation or missing realization of those characteristics is deemed frustrating by workers. For this, the average rating scores were calculated based on the workers' assessments on the 5-point Likert scales. Figure 2.7 shows the corresponding means of both facets plotted against each other for both platforms. While the importance of the different categories regarding the workers' choice of tasks is displayed on each x-axis, the degree of frustration (on the y-axes) increases with the dissatisfaction over certain features, or even their absence. Hence, each task property results in one data point within the figures. A point on the line indicates that the extent to which the corresponding aspect is important for task selection is the same as the extent to which it is deemed frustrating when being present in a negative form.

First, a closer look is taken at the aspects workers from Microworkers consider to be important for the task selection, shown in Figure 2.7a. The most



(a) Ratings from workers from Microworkers.



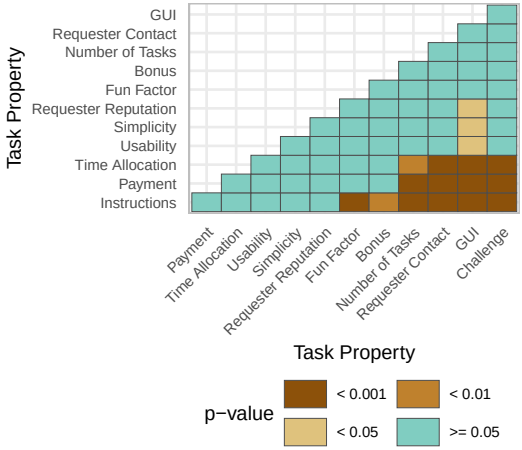
(b) Ratings from users from MTurk.

Figure 2.7: Comparison of important and frustrating aspects for selecting tasks.

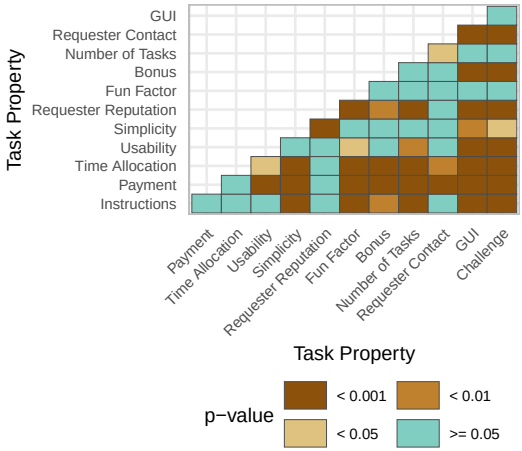
important characteristics are clear instructions, the reward, and the estimated time to complete the task followed by the simplicity and the usability of the interface. An appealing graphical design is rated worst. Friedman's test shows that there are significant differences in the ratings of the importance of the aspects ($\chi^2(11) = 131.34$, $p < 0.001$). The ratings are further analyzed with the pairwise comparison using Nemenyi's multiple comparison test. The significance of the differences is shown in Figure 2.8a. As indicated by Figure 2.7a the top three important aspects are significantly more important than the four properties that are rated as least important. However, there is no significant difference between ratings of clear instructions, appropriate payment, the required completion time, the usability of the task interface, and the simplicity of tasks.

Next, aspects which are considered as important during the task selection on MTurk are analyzed (cf. Figure 2.7b). The three most important task properties on MTurk are identical to the ones on Microworkers, but with a different ordering. Again, a user interface with an appealing graphical design is least important for workers from MTurk. Friedman's test results again in significant differences in the ratings of the properties ($\chi^2(11) = 354.05$, $p < 0.001$). The results of Nemenyi's pairwise comparison is presented in Figure 2.8b. While the differences between the four most important aspects are not significant, they are significantly more considered during the task selection than most of the other aspects.

The findings concerning the aspects within the upper three positions are only partially in line with the results from Schulze et al. [24]. While the reward and the task description were also among the top three aspects identified as important by the scientists, tasks that sound interesting or enjoyable were rated to the first rank in the context of their survey. Further, the finding indicates that an easy-to-use page design (usability) plays a major role during task selection and that workers place more value in it than in having fun performing the task or in the ability to work on multiple tasks of this kind. However, comparing the results of users from Microworkers with users of MTurk, the results differ slightly concerning the significance and the order of the results. While certain properties



(a) Significance of importance ratings on Microworkers.



(b) Significance of importance ratings on MTurk.

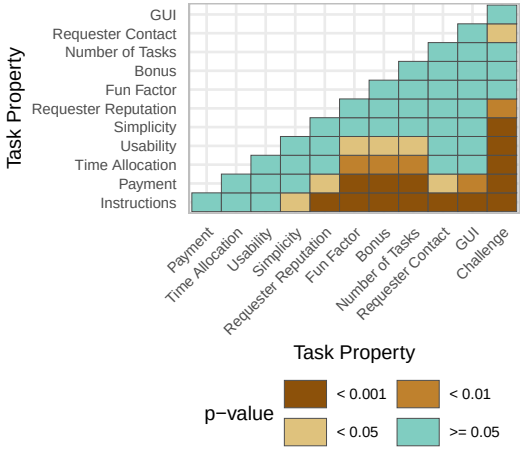
Figure 2.8: Significance of rating differences of the importance of task properties on both platforms.

can be clearly identified as most important on MTurk, there are only tendencies which are the most significant factors while selecting tasks on Microworkers.

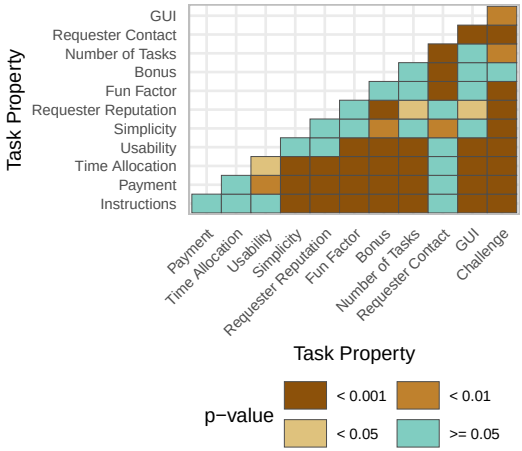
The workers were furthermore shown the negative formulations of task properties and had to rate the extent to which they deem them frustrating, shown on the y-axes of Figure 2.7. At first glance, there are quite a few similarities between the results of frustrating and important features, including the fact that the aspects related to the payment, the estimated completion time, and the instructions were rated highest again. The usability is in the forth place on Microworkers and in the fifth place on MTurk. While on the importance scale an appealing interface was rated lowest, the least frustrating aspect is a lack of challenge while working on tasks.

By using Friedman's test the significance of these ratings is analyzed once more. It shows for ratings on Microworkers ($\chi^2(11) = 160.42$, $p < 0.001$) as well as for answers given by users from MTurk ($\chi^2(11) = 467.75$, $p < 0.001$) that the ratings differ significantly. An overview about the significance resulting from a pairwise comparison of the task properties is given in Figure 2.9. On Microworkers, there are again no significant differences between the four aspects rated as most frustrating. However, ambiguous instructions are perceived as significantly more frustrating than the remaining aspects. Further, a bad usability is rated as more frustrating than boring tasks, no bonuses, the availability of only a few tasks of the same task type, and a missing challenge while working on tasks. The analysis of the ratings from MTurk shows significant differences between ambiguous instructions and all other aspects, except the other top three rated properties and the missing possibility to contact the requester. Further, an inadequate payment and miscalculated time needed to complete the task are significantly more frustrating than a user interface with bad usability.

Overall, the following observations can be made. First, the results show once more the importance of the payment-related factors, including the reward and the allocated completion time, and the importance of good task instructions across both platforms. Besides this finding, there are differences between the opinion of the workers from different platforms concerning the importance of



(a) Significance of frustration ratings on Microworkers.



(b) Significance of frustration ratings on MTurk.

Figure 2.9: Significance of rating differences on the frustration about the absence of task properties on both platforms.

other task properties. While the perception of users from MTurk is more consistent, resulting in ratings with significant differences, ratings of workers from Microworkers are more often inconsistent. Second, the results are only partly in line with findings of previous studies, e.g., by Schulze and colleagues [24]. Third, the usability, which was not considered in previous studies, obtained an average rating score almost identical to the one of the reverse formulation on MTurk and only a slightly lower one on Microworkers. This suggests that a good usability is not only deemed important for task selection but that it is also perceived as frustrating if the ease of use of a page is low. For the remaining categories except the one regarding the possibility to contact the requester on MTurk, the corresponding data points are below the line. This can be seen as an indicator that workers consider them as more or less important for the task selection, but do not perceive it as frustrating to the same extent if the respective property is missing or poorly implemented. This is especially true for the challenge of the task, the bonus for a good performance, the reputation of the requester on MTurk, and the fun factor on Microworkers.

Design Preferences

Apart from examining the importance of a good usability and the user interface design from the workers' perspective, the study also aimed to gather their opinion on certain design aspects and analyze some specific task selection habits. The results of the corresponding questions are shown in Figure 2.10. The y-axis displays the different aspects of investigation, the x-axis shows how often which of the two predefined, mostly contrary options, was selected.

Overall, the results indicate that there is a consensus on some of the aspects between the workers within the platforms, e.g., on the usefulness of illustrations and highlighting. Consensus means that at least 75 % of the participants within a platform selected the same option. The number of aspects with consensus is higher between the workers from Microworkers (seven properties) than on MTurk (four aspects).

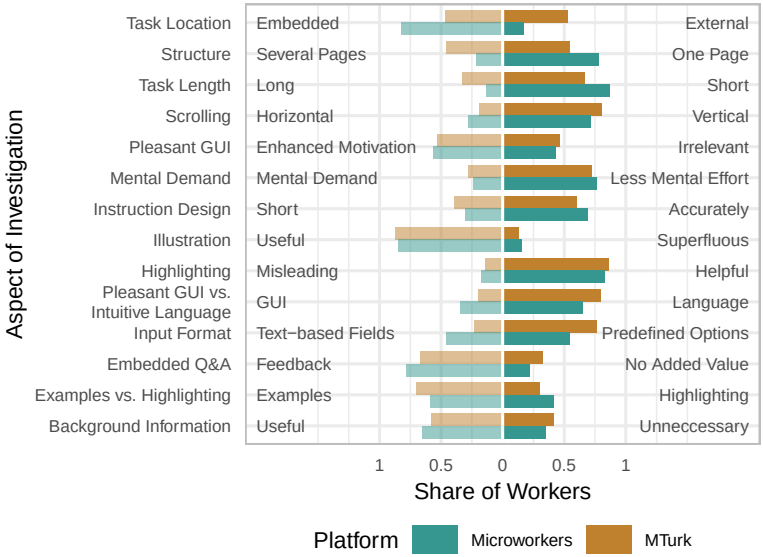


Figure 2.10: Design preferences.

Nevertheless, the opinion of the workers also often varies concerning the design preferences, e.g., including examples in the instructions or highlighting important parts of the instructions on Microworkers, and the usefulness of background information about the task on MTurk. A varying opinion means that only 40 % to 60 % of the participants have chosen the same answer. Especially, the participants are divided concerning the motivation of a pleasant user interface, with 53.2 % on MTurk and 56.5 % on Microworkers stating that it increases their motivation to work on the given task and hence has a positive influence on the performance. Finally, it should also be recognized that when the workers had the option to choose between an appealing user interface and an intuitive instruction language, 79 % on MTurk and 65 % on Microworkers would prefer

an comprehensible language over a pleasant GUI. This result is in line with the rating of the importance of clear instructions and an appealing user interface from the first part of the survey.

Comparing the results between the platforms by using Pearson's chi-squared test with Yates' continuity correction for each design aspect, significant differences between the opinions of the workers from both platforms concerning the task location ($\chi^2(1) = 14.87$, $p < 0.001$), the structure ($\chi^2(1) = 6.58$, $p < 0.05$), the task length ($\chi^2(1) = 5.33$, $p < 0.05$), and the input format ($\chi^2(1) = 14.87$, $p < 0.001$) were found. This indicates that workers from different platforms have different opinions concerning their preferred task design. While workers from Microworkers favor short tasks presented on a single page embedded into the platform, the users of MTurk have no clear accordance concerning the location, the number of pages, and the task length. But they are in accordance about preferring predefined options as input format. The differences across the platforms may be caused by platform-specific characteristics, e.g., the creation process of tasks and predefined task templates and designs. The varying preferences within the platforms may be caused by worker characteristics, e.g., preferred kind of tasks. To further analyze this aspect, a cluster analysis of design preferences is performed to cluster the workers based on the self-reported demographic information and working behavior on the platforms. For the clustering a hierarchical approach and the partitioning around medoids (PAM) algorithm based on Gower dissimilarities for categorical data is used. However, no clear subgroups could be identified with consistent preferences.

In conclusion, the overall result encourages to pay further attention to the differences between crowdsourcing platforms and to the interface design of crowdsourcing tasks. There is clear indication that the usability – which is substantially influenced by the interface design – does not only influence the task complexity and hence the workers' performance as outlined in Section 2.1.1, but is also important from the workers' point of view concerning the task selection.

2.4 Impact of Task Decomposition

As the complexity of crowdsourcing tasks is important for the task selection by the workers, see Section 2.3, reducing the complexity may positively influence the completion time of tasks and the quality of work results. A common used approach for simplifying complex tasks is their decomposition to sub-tasks [56, 57]. Even if studies indicate that there might be a relation between the quality of work results and the degree of decomposition [39, 40], there is still no quantitative study focusing on the impact of task decomposition on the quality of work results.

This section aims to fill this gap by analyzing the impact of decomposing tasks on the quality of work results with a user study. Furthermore, the study also considers side effects caused by the workers' familiarity with the task content. Being familiar with the content may result in higher quality as it makes it easier to deal with complex not decomposed tasks.

2.4.1 Study Design and Conduction

The impact of the decomposition is analyzed for a typical task on microtasking platforms in the area of data engineering – the extraction and completion of data. The participants of the study have to extract information of scientific references formated with BibTeX[2]. Each presented reference has the same structure:

Authors: Title. Conference Information. Address, Year.

The job of the workers is first to extract the authors and the title of the publication as well as the information about the conference, i.e., name, address and date (year), second they have to search the Web and complete the full names of the authors. For all publications, only the initials of the first names are given.

[2] http://www.bibtex.org/de/; Accessed: August 1st, 2020

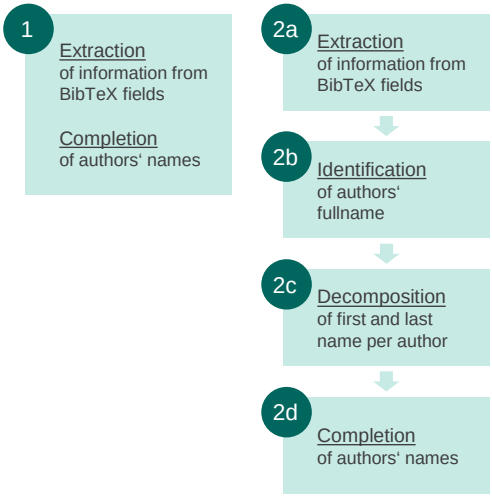


Figure 2.11: Task content with and without decomposition.

This kind of task offers the opportunity to decompose it into sub-tasks of different complexity and content, i.e., simple information extraction vs. searching the Web for information. Further, the results are easy to verify.

To evaluate the impact of the decomposition, the task is split into four sub-tasks, schematically depicted in Figure 2.11. While Task 1 contains both, the data extraction and completion, in Sub-task 2a-d these steps are decomposed. The extraction of information is still done in one step (Sub-task 2a). Next, the authors are separated (Sub-task 2b) as well as the first and last name of each author (Sub-task 2c). In the last step, the workers only have to complete the first names (Sub-task 2d).

These (sub-)tasks are structured as follows. First, the instruction is shown including an example. Then the participant has to complete a training task for which the input is verified automatically. If the worker has made mistakes, these

Table 2.5: Overview about study setup.

Task	References	Length [min]	Reward [$]	Participants
Task 1	5	15-20	0.40	96
Sub-task 2a	10	8-10	0.10	36
Sub-task 2b	10	3-5	0.10	49
Sub-task 2c	10	3-5	0.10	48
Sub-task 2d	10	8-12	0.15	37

are marked and the correct solution is presented. After completing the main (sub-)tasks (screenshots of the user interfaces are in Appendix B.3), a final questionnaire is shown. Here, demographic information as well as information about the familiarity of the worker with the BibTeX format of references are collected.

The study was fielded on Microworkers from February to March, 2016. The tasks were available to all workers from Microworkers. Table 2.5 gives an overview about the study setup. Each task without decomposition (Task 1) contains five references which the workers have to process. These references are randomly selected out of 100 scientific publications. The publications were manually reviewed and the missing information were completed by using the ACM Digital Library[3] and Google Search[4]. As the sub-tasks are less complex, more publications need to be processed, i.e., ten references. Furthermore, the estimated completion time varies as well as the reward payed for successfully submitted results based on the task length and complexity.

Overall, 266 workers passed the consistency checks based on the self-reported information and information provided by their public profile on Microworkers. The study participants processed up to 2 370 references. The analysis of the self-reported demographic information of the workers shows that more than half of the participants are from Bangladesh, followed by Serbia and India. Two of these has been top-countries of residency of users from Microworkers in 2016 [74].

[3] https://dl.acm.org/; Accessed: August 1st, 2020
[4] https://www.google.com/; Accessed: August 1st, 2020

Table 2.6: Demographics of the participants.

Category	Value	Task 1	Sub-task 2a-d
Gender	Female	16 (16.7 %)	20 (11.8 %)
	Male	78 (81.2 %)	149 (87.6 %)
	Not specified	2 (2.1 %)	1 (0.6 %)
Age	< 18	3 (3.1 %)	3 (1.8 %)
	18–30	81 (84.5 %)	133 (78.2 %)
	31–50	11 (11.4 %)	33 (19.4 %)
	> 50	1 (1.0 %)	1 (0.6 %)
Familiar with BibTeX	Yes	28 (29.2 %)	52 (30.6 %)
	No	67 (69.8 %)	118 (69.4 %)
	Not specified	1 (1.0 %)	-

Table 2.6 gives an overview about the demographics of the workers for Task 1 and Sub-task 2a-d. Most of the workers are male and in the age between 18 and 30 years. About 69 % of the participants of each task type answered that they are not familiar with BibTeX. Using Pearson's chi-squared test, it is analyzed if the samples of each demographic category originate from the same distribution. All tests are not significant, thus, the null-hypotheses can not be rejected (age: $p = 0.347$, gender: $p = 0.273$, familiar with BibTeX: $p = 0.404$). This indicates, that there are no differences in the distributions of the workers' characteristics for both task types.

2.4.2 Impact on Work Performance

To evaluate the impact of the task decomposition, the quality of the submitted results is analyzed. The quality is defined by the normalized Levenshtein's editing distance. To recap, Levenshtein's distance describes the number of insert, delete and transform operations needed to convert one string into another. The distance is normalized by dividing the distance by the length of the longer string. Thus, the quality value is in a range between 0 and 1. The smaller the

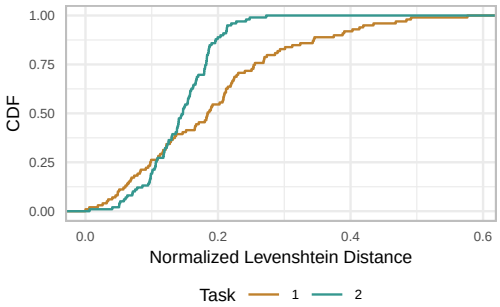


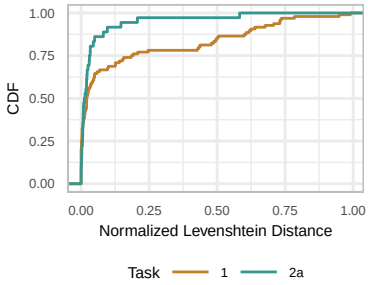
Figure 2.12: Cumulative distribution function of the average normalized Levenshtein distance of expected and submitted results of the references for each task type.

value, the less operations are necessary to convert the submitted strings to the correct solutions, meaning less mistakes have been made.

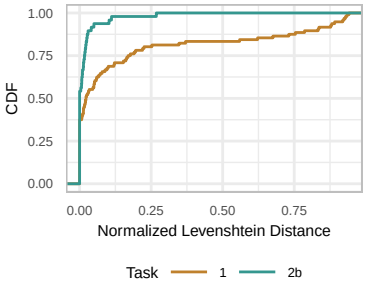
First, the result per reference is analyzed to compare the overall performance of the workflow with and without decomposition. Figure 2.12 shows the cumulative distribution function of the average quality the references for both workflows. The more the curve is to the left, the higher the quality of the results. Comparing the results of Task 1 and Sub-tasks 2a-d, the quality of the combined results of all sub-tasks is higher than for Task 1. As the samples are neither normal distributed nor the variances are homogeneous, the origination of the samples from the same population is examined with Pearson's chi-squared test. The decomposition has a significant effect on the quality ($\chi^2(13) = 49.41$, $p < 0.001$) with an average normalized Levenshtein distance of 0.194 for Task 1 and 0.144 for Sub-tasks 2a-d. For quantifying the size of the effect, Cramer's V is used which is a normalized value between 0 and 1 based on the χ^2-value. Cramer's V shows a positive effect of 0.499. Reasons for the higher quality may be the predefined strategy to solve the task. Hence, the overall task is more

structured and more clear. Further, poor results of sub-tasks only partly affect the overall working quality of the references.

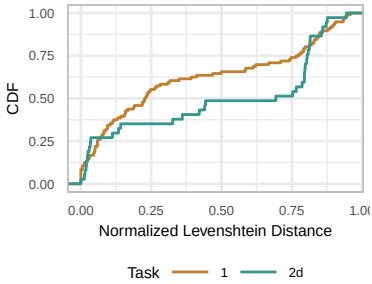
By analyzing the results per sub-task, i.e., for the extraction of the title, conference, place, and year as well as the authors' last names, and the completion of the first names, the impact on the different working steps is evaluated. Figure 2.13 shows the cumulative distribution function of the average, normalized Levenshtein distance for the workers for each sub-task and the associated working step of the task without decomposition. The results for title, conference, place, and year are presented in Figure 2.13a. Even if slightly more workers achieve results of higher quality when working on Sub-task 2a, the decomposition has no significant impact on the quality ($\chi^2(11) = 15.038$, $p = 0.187$) with a median normalized distance of 0.021 for Task 1 and 0.013 for Sub-task 2a. The seen effect is of medium size with Cramer's $V = 0.337$. As the extraction of the information is obviously the first step to solve the task and it is explicitly described in both instructions, the result is as expected. However, the decomposition has a significant effect on the extraction and completion of the authors' full names. The quality of last names is positively affected, cf. Figure 2.13b, indicated by Pearson's chi-squared test ($\chi^2(13) = 22.768$, $p < 0.05$). The effect is medium sized with Cramer's $V = 0.398$. Reason for the positive effect may be the simplicity of the working step requested in Sub-task 2c, resulting in outcomes without mistakes. The median distance for Sub-task 2c is 0.00 whereas the median in Task 1 is slightly larger with 0.02. Workers processing Sub-task 2c only have to identify the last names in given full names. The full names were already extracted and separated for each author in the previous sub-tasks. Finally, the effect is reversed for the completion of the first names, shown in Figure 2.13c. The median distance is larger for the decomposed Sub-task 2d with 0.693 than for Task 1 with 0.229. The negative impact of the decomposition is significant ($\chi^2(13) = 26.423$, $p < 0.05$), with a medium to large sized effect indicated by Cramer's $V = 0.446$. Thus, a negative effect is seen when separating the two task types, i.e., data extraction and completion, which may lead to loose the context of this sub-task. Further, the completion of the first names is more



(a) Extraction of title, conference, place and year.



(b) Extraction of authors' last name.



(c) Completion of authors' first name.

Figure 2.13: Cumulative distribution function of the Levenshtein distance of the results per sub-task, i.e., BibTeX fields.

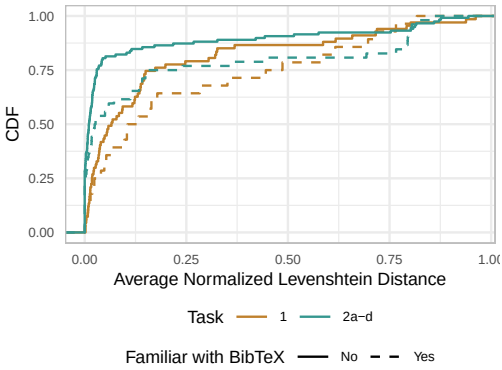


Figure 2.14: Cumulative distribution function of normalized Levenshtein distance of the submitted results for workers who are familiar and not familiar with the BibTeX format.

complex than the extraction of information. While the extraction process is decomposed in Sub-task 2a-c, the completion step is not further split up to reduce its complexity.

By comparing the results of the workers who are familiar with the BibTeX format and of those who are unexperienced, the impact of previous knowledge about the content of the task on the quality of the submitted results is analyzed. The cumulative distribution function of the average normalized editing distance for experienced and unexperienced workers with BibTeX is shown in Figure 2.14. Other than expected, the comparison of the distributions for both task types indicates a trend towards results with lower quality for workers who are familiar with BibTeX. The difference is not significant within the group of workers processing Task 1, established by Pearson's chi-squared test ($\chi^2(9) = 5.092$, $p = 0.826$) with a small sized effect (Cramer's V $= 0.139$). Within the workers of Sub-tasks 2a-d the effect is significant ($\chi^2(13) = 27.281$, $p < 0.05$) and medium to large (Cramer's V $= 0.321$). A similar, not significant effect is seen

for experienced and unexperienced workers across the tasks ($\chi^2(13) = 14.699$, $p = 0.326$). This effect is small to medium (Cramer's V $= 0.235$). The unexpected trend to results of higher quality from workers who are unfamiliar with the task subject may be caused by side effects which the study does not measure. Another reason may be that the familiarity is self-reported and thus may be error prone. Experienced workers may state that they are unfamiliar expecting that mistakes are rated more tolerant by the employer. Conversely, workers may report to be an expert even if they are unfamiliar to be invited to an expert group for such kind of tasks in the future. Overall, the results are in line with the findings of Winther et al. [84] that gives evidence that unexperienced workers achieve comparable results to experienced workers when providing them clear instructions including examples and training tasks.

To sum up, this study gives evidence that the decomposition of tasks may improve the overall task quality. The detailed analysis of the results of the decomposed sub-tasks shows that this improvement is not true for all kind of subtasks. Splitting up the task into working steps of different task types, i.e., data extraction and completion, only increased the quality of the extraction step. This may be caused by the fact that the data extraction step is again decomposed to sub-tasks while the data completion sub-task is only separated from the information extraction process, but is not split up further. Even if the study is limited to two kind of tasks, the results indicate that it is not always the best approach to decompose a task to its maximum due to the risk of loosing the context of the single steps. Instead, providing good instructions including examples and training phases may be the more promising and efficient approach.

2.5 Impact of Working and Instruction Language

As the clarity of task instructions is an essential factor for the task selection and performance of the workers, employers should especially put attention on the instruction while creating tasks. Besides the instruction's structure and presentation, the used language itself plays an important role to ease the understand-

ing of the workers and to prevent misunderstandings. Providing the instructions in native language to the workers may reduce language barriers, improve the quality of work results, and reduce the time needed to complete tasks. Furthermore, working in native language, e.g., in tasks like writing articles, comments, or simple tagging images, may also improve the outcome due to a better feeling for the language. Offering this opportunity to the workers would require the translation of the instructions to the native language of the workers and the translation of the submitted task results to English or another target language. The translation could be done via crowdsourcing or to reduce the costs by using translation tools which can be used for free.

In this section the influence of the instruction and working language is investigated. For the influence of the language of instructions, the quality of work results of tasks providing instructions in mother tongue and in English, which is the common language on microtasking platforms, is analyzed. The evaluation of the impact of the working language is based on the quality of work results produced in mother tongue and submissions in English.

2.5.1 Methodology

The impact of the language on the quality of task results is evaluated by conducting two separated user studies on the crowdsourcing platform Microworkers. In the first study, the impact of the language of instruction is investigated by using the attentiveWeb test, as this task is not self explanatory. Second, the impact of the working language is analyzed based on the results of a typical crowdsourcing task – image tagging.

Study Design - Language of Instructions

For the evaluation of the impact of the language of instruction on outcome's quality the attentiveWeb application is used. The attentiveWeb test is an online tool for measuring the concentration, described in detail in Section 2.2.2. This task is chosen as it is simple, but not self explanatory at all. In addition, by using

standardized metrics it is easy to identify workers who do not work seriously. An indicator for misunderstanding the task is the ratio of workers who have a minimum concentration performance less or equal to zero in combination with an error rate of more than 100 %. Another indicator for misunderstanding the instructions is the drop out rate, meaning the ratio of workers who leave the task after reading the instructions. Besides effects on the understanding, the attention of the workers may be also influenced when working in a foreign language. A decreased attention may negatively influence the quality of work results, as shown in Section 2.2.3.

The task description, including the welcome message, a short description of the task context, and the test instruction is shown either in English or in mother tongue. The language is set randomly for each worker entering the landing page of the study. After reading the test instructions, the processing of the test is language independent.

Study Design - Working Language

A simple image tagging task is used to evaluate not only the impact of the language of instructions but also investigate the quality of the results when working in native language. Such an approach can be realized by translating the task results submitted by the workers into the desired target language with translation tools such as Google Translate[5] or Bing Microsoft Translator[6]. These tools are free to use and they often provide APIs. Hence, this approach can be automated with no additional costs.

The task is designed as follows. First, the instructions are shown to the study participants. The instructions are presented in English or in the worker's native language. To prevent misunderstandings due to mistranslation, the English version of the instructions is additionally shown to the instructions in mother tongue. Depending on the language of instructions the workers are explicitly asked to work in English or in their native language. Again, the language is

[5] https://translate.google.com; Accessed: August 1st, 2020
[6] https://www.bing.com/translator; Accessed: August 1st, 2020

randomly chosen for each worker when accessing the study. After reading the instructions, the workers have to tag five images taken from pixabay[7]. Each image requires three tags which should describe its content or context.

Conduction of Studies

The studies were conducted with users from Bangladesh, as it was one of the top countries of Microworkers in 2018 [74]. In Bangladesh Bengali is the local language spoken from up to 98 % of the population[8]. The estimated completion time is 10 min with a compensation of US$0.10 for the image tagging task and 20 min with a payment of US$0.20 for the attention test.

Before conducting the studies, the instructions are translated to Bengali in a separate task on Microworkers using the workflow shown in Figure 2.15. After

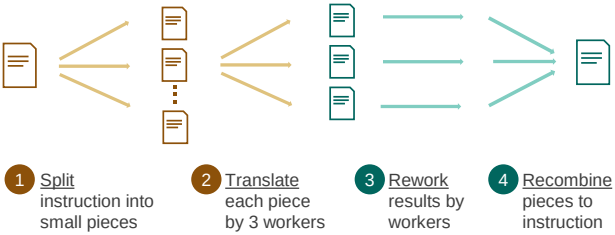


Figure 2.15: Schematic process of translating instructions into native language.

breaking down the instructions into smaller parts and sentences (Step 1), these are each translated by three workers (Step 2). If the translations are identically, it is assumed that this part of the instruction is translated correctly. Otherwise, the three versions are revised by other workers who are asked to select and

[7] https://pixabay.com; Accessed: August 1st, 2020
[8] https://worldpopulationreview.com/languages/bangladesh; Accessed: August 1st, 2020

improve the most suited version (Step 3). Finally, the versions are recombined to one document (Step 4).

In total, 90 workers finished the image tagging task in April 2018, 45 workers working in English and 45 participants working in their native language, respectively. They submitted in total 1 365 tags, 690 in English and 675 in Bengali.

The attention test was conducted from August to September 2018. Overall, 320 workers (140 with the instructions in English) started the attention test. Of those, 193 participants (89 with instructions in English) completed the study.

2.5.2 Evaluation

In this section, first, the impact of the language of instructions on the quality of work results is evaluated. Second, the results of the study about the impact of the working language are presented.

Impact of Language of Instructions

The analysis of the impact of the language of instructions on the quality of work results, is done in two steps. First, the share of workers passing several filter criteria, i.e., reading the brief introduction to the study, continuing the test after reading the instructions, processing each test page, achieving a concentration performance larger than zero in combination with an error rate below 100 %, for both test groups is evaluated. Second, the attentiveness of workers passing the minimum filter criterion for the attention test attentiveWeb is analyzed as the attention of the workers may be affected by the language of instructions. An decrease of attention may lead to lower results as the attention and the working performance are correlated, as discussed in Section 2.2.3.

The considered filters are presented in Figure 2.16. The filters are applied to all unique participants who visited the study, regardless of whether they completed or aborted the task. *F1: Test started* leaves participants who visit at least the first test page. Leaving the task after reading the instructions may be an indicator that the worker did not understand the required work. *F2: Test completed* leaves

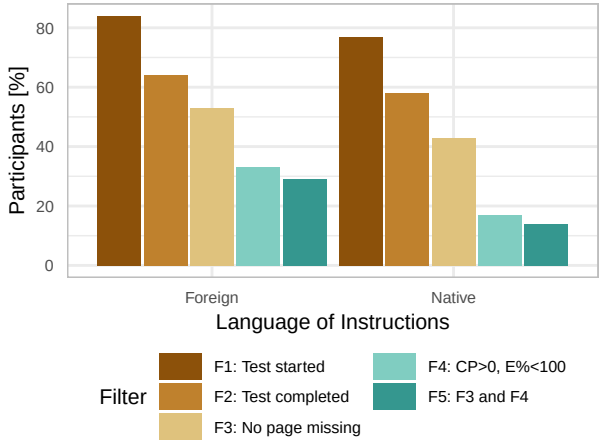


Figure 2.16: Percentage of participants for both languages of instructions passing the filters F1: starting the attention test after reading the instructions, F2: completing the test, F3: process all test pages, F4: having a concentration performance larger than zero and an error rate below 100 %, F5: the combination of F3 and F4.

participants who submitted the final questionnaire. The filter *F3: No page missing* leaves workers who clicked at least one button per test page. *F4: CP > 0 and E% < 100* leaves participants who clicked more target items than non-target items in the attention test. Conversely, clicking more non-target items would indicate that the participant misunderstand or boycott the test. *F5: F3 and F4* combines the two former described filters.

The evaluation of filtering results shows that the share of workers passing the filters *F1*, *F2*, and *F3* is slightly lower for the instructions in native language (Bengali). Pearson's chi-squared tests with Yates' continuity correction result in no significant differences in the sample distributions (*F1*: $p = 0.121$, *F2*: $p = 0.349$, *F3*: $p = 0.174$). At a first glace, providing the instructions in the workers' native language has neither a positive nor a negative effect on the understanding of the

workers. This observation is strengthened by the results of filter *F3* which analyzes the behavior during the test. A different result is seen for *F4* leaving participants based on their test performance. The homogeneity test establishes that the samples do not originate from the same population ($\chi^2(1) = 9.697, p < 0.01$). Less workers passing the performance based filter when reading the instruction in their native language. This is an indicator, that these workers did not understand or boycott the test. The effect is of small size (Cramer's V$= 0.174$). A similar result is seen for the evaluation of filter *F5* ($\chi^2(1) = 9.601, p < 0.01$, Cramer's V$= 0.173$). This observation should be used only as an indicator that there is a relationship between the language of instructions and the test performance, as it is unclear if the workers did not understand or boycott the test. Additionally, the results may be also caused by ambiguities in the instructions resulting from the translation.

As the language of instructions may influence the worker's concentration performance and finally the quality of working results, this relation is evaluated next. On average, participants who get the instructions in foreign language achieve a higher CP ($mean = 95.78$, $SE = 6.08$), than workers reading the instructions in mother tongue ($mean = 76.92$, $SE = 10.20$). Welch's t-test results in no significant difference ($t(42.57) = 1.59, p > 0.05$), with a small effect size of $r = 0.24$.

Regarding the processing accuracy of the test, average E% is lower for participants with the English instructions ($mean = 33.85$, $SE = 3.84$), than for workers with instructions in native language ($mean = 45.86$, $SE = 6.32$). This difference is also not significant, shown by the results of Welch's t-test ($t(43.19) = -1.62, p > 0.05$). The effect size again is small with $r = 0.24$.

The results show, other than expected, that the language of instruction has neither a positive nor a negative effect on the workers before starting the task. Nevertheless, there is an indicator for a negative effect while processing the test. Instructions in native language lead to more workers who clicked more non-target items than target items. Reasons for this may be mistakes or misleading parts in the instructions originated from the translation process or due

to differences in the meaning of words for different dialects in Bangladesh. On the other side, the provided English description of the test is very simple. Thus, it is easy to understand, especially for workers who are used to work in English. However, providing the task description in native language has no effect on the attention of the workers.

Impact of Working Language

For analyzing the impact of the working language, the results of the image tagging task are evaluated. To do so, the submitted tags in Bengali are translated automatically to English using the translation tool Google Translate. The quality of the translated tags and the tags submitted directly in English are classified manually by an expert. The expert distinguishes between tags describing the content or context of the image, classified as usable, and tags which are off topic and therefore non-usable. Furthermore, assessments of the image or whole sentences describing the image are classified also as non-usable. As it is unclear, if not translatable tags are non-usable due to a missing translation or due to misspelling, they are marked as non-translatable, in addition. Tags with English as working language are also marked as non-translatable if they are no correct English words.

Overall, only 2.1 % of the Bengali tags are non-translatable, while 14.3 % of the English tags are not correct, e.g., due to misspelling. This difference is significant with $\chi^2(1) = 67.695$, $p < 0.001$. Further, working in native language has a positive effect on the share of usable tags (Bengali: 78.4 %, English: 64.5 %) which is significant ($\chi^2(1) = 32.150$, $p < 0.001$).

This positive effect is also seen for each image. Table 2.7 shows that for each image the share of usable tags is significant higher while working in native language than working in English. Other factors which may have positive effects on the working performance may be the amount of time workers already spent on the platform indicated by the overall time they are registered on the platform and the total amount of tasks done. One can assume that after completing a cer-

Table 2.7: Share of usable tags per image per working language.

Language	Mean	SE	CI
English	0.643	0.025	±0.052
Bengali	0.777	0.027	±0.055

tain amount of tasks, e.g., 100 tasks, the workers get used to work in English. An ANOVA test did not establish this hypothesis ($p > 0.05$).

Next, the diversity of the created tags is evaluated. One would expect that working in native language leads to a more diverse set of tags. To evaluate this hypothesis, the term frequency (tf) adjusted for document length is calculated, inspired by the work of Jiang et al. [66]. The set of tags for each working language defines a corpus, while the tags per image represent the documents. The tags submitted in Bengali are analyzed without translation, to avoid biases. Furthermore, non-usable and non-translatable tags are omitted, resulting in 364 tags in native language and 446 tags in English. Of those 42.6 % are unique in mother tongue and 34.7 % are submitted only once in foreign language.

Figure 2.17 shows the cumulative distribution function of tf for the distinct tags. A small value of tf means that a tag is submitted rarely for the related image. Thus, a curve in the upper left represents a corpus with a large amount of tags submitted only once per image, indicating a diverse set of tags. Even if the two sample Kolmogorov-Smirnov test results in a rejection of the null-hypothesis that both samples originate from the same distribution ($D = 0.227$, $p < 0.001$), there are only some minor differences between tfs for foreign and native language. On the one hand, more tags in native language are submitted less frequent than in English. On the other hand, parts of the tags are submitted more frequent than tags produced in English. Overall, the analysis shows that the language has no large effect on the diversity.

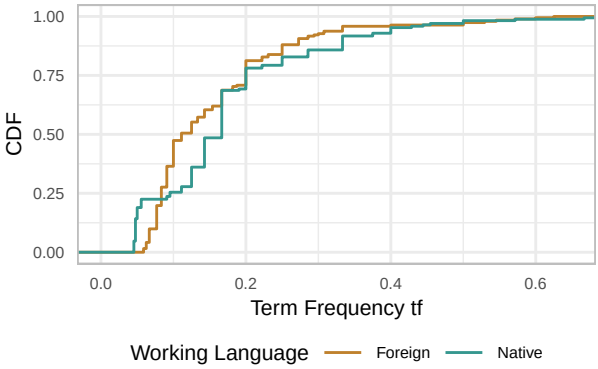


Figure 2.17: Cumulative distribution function of term frequencies tf of tags created in foreign and native language.

2.6 Lessons Learned

This chapter discussed the necessity of ensuring quality in the context of crowdsourced tasks considering both the employers' and workers' point of view. Furthermore, it investigated an attention-based approach for quality control as well as approaches for reducing the complexity of tasks. These optimization approaches consider workers' preferences and important factors while selecting tasks. Hence, the gap in literature as outlined in Section 2.1 is closed.

Section 2.2 introduced the attention test attentiveWeb and investigated the applicability of the test on crowdsourcing platforms by comparing the attentiveness of crowd workers and users from an online panel used for psychological studies. Even if there are differences in the attention of the users from the panel and the workers from two crowdsourcing platforms, the test is applicable and usable for identifying attentive workers providing high-quality results. The predictive and filter validity is evaluated through additional user studies. The results showed that the quality of work results and the workers' attention were correlated. Nevertheless, filtering workers based on their attention also leads to

an exclusion of reliable workers. This effect may occur due to the mechanism's independence from the task content.

As quality control mechanisms often only consider the employers' point of view, in Section 2.3, important factors and design preferences of the workers while selecting and performing tasks were investigated. The results are partly in line with previous studies on important task properties, confirming the importance of extrinsic motivational factors, i.e., appropriate payment and allocated time for completing tasks, and clear task instructions. However, providing a user interface with good usability is almost as important as the fun factor of tasks which are new findings. While workers from different platforms perceived similar task properties as important during the task selection, their preferences concerning the task design differed. Even if there is consensus with respect to some of the design properties, the workers disagreed in multiple aspects. Thus, it is nearly impossible to design a task that meets all preferences of all workers.

Considering workers' perspective based on the results of the study, Section 2.4 and Section 2.5 presented two approaches to reduce the complexity of tasks, and thus, improve the results' quality. The decomposition of tasks including different kind of sub-tasks increased the quality of the results only to a certain extent. While sub-tasks which requested the extraction of information benefited from the decomposition, more sophisticated sub-tasks, e.g., completion of data by searching the Web, suffered. Thus, decomposing a task to its maximum does not necessarily increase the outcomes' quality as it may result in loosing the context of sub-tasks.

A more promising approach for improving the task quality is providing task instructions in the workers' native language. Even if the user study, presented in Section 2.5 did not give evidence on a positive effect, enhancing the approach by allowing the crowd to work in their mother tongue while producing text, e.g., tagging images, improves the results. Besides this positive effect, the studies also showed that good instructions reduce side effects, which would affect the workers' performance negatively. Conversely, bad instructions dominate other negative factors.

3 Monitoring QoE of Enterprise Applications

Due to globalization and digitization of processes in a wide range of businesses, business software, systems, and services found their way into the day-to-day work of the employees. In most companies, enterprise applications, such as office products or databases, are heavily used during work hours. Outages or slow response times of the systems not only slow down business processes, but might also increase the frustration of the workforce. This becomes even more important as an increasing number of business applications are served remotely from a server, e.g., in a data center or a cloud. This introduces additional network delays depending on the amount of transferred data, the physical location of the server, and the network capacity and load. Thus, the employees' satisfaction with the system performance has to be considered as a major business driver, as it directly influences the motivation and productivity of the employees [85].

Assessing the performance of applications and systems based on Quality of Service (QoS) parameters, e.g., network delays or packet loss, is no longer sufficient as it does not include the user's perception. In this context, the concept of Quality of Experience (QoE) arose, which describes the degree of delight or annoyance of the user with the quality of an application or service [18]. Here, the individual perception may be influenced by multiple factors and differs depending on the application's characteristics as well as on the context of usage.

While QoE and its influence factors have been widely studied for personal multimedia applications, such as video streaming or VoIP telephony, the application of the concept to the business usage domain is still in its infancy.

Common approaches to investigate influence factors, i.e., classical lab or crowdsourced QoE studies, can hardly be transferred to the business domain as such studies would influence the daily business of the company and new challenges arise. First, the IT infrastructure may be highly complex, which makes it hard to identify the most important technical parameters of the business application. Second, the company might be reluctant to monitor or influence a production system. Third, as the QoE of enterprise applications can only be meaningfully assessed in the work context, the employees would need to participate in studies during working, which would be time consuming and potentially distracting, both negatively affecting the performance of the workforce.

Adapting existing methodologies and standards for QoE tests to the domain of business applications is accompanied by several research questions. In the first place, the question is how to monitor QoE in enterprise environments considering enterprise specific requirements and follow standards for QoE monitoring? Related to the evaluation of influence factors, questions, such as how to collect quality assessments of business applications in a valid and representative way, are relevant.

Figure 3.1 gives a brief overview about the research questions addressed in this chapter and the methodology used to answer them. Besides proposing a concept for monitoring QoE in enterprise environments, the chapter focuses on QoE studies for business applications, meaning the collection of performance ratings from employees. Here, relevant dimensions of the study design are discussed and it is investigated how the dimension of system artificiality and the domain knowledge of study participants influence the validity and representativeness of the monitored QoE. Finally, a tool for conducting QoE studies in enterprise environments which considers enterprise specific requirements is introduced. The applicability of this tool and its acceptance by the employees are analyzed and discussed based on studies conducted in a large, cooperating company.

The remainder of this chapter is structured as follows. A brief introduction into the concept of QoE is given in Section 3.1. Afterwards, approaches for mon-

Section	Research question	Methodology/Contribution
3.2	How to monitor QoE in enterprise environments?	Presentation of a concept based on literature search and interviews with experts from a cooperating company
3.3.1	What are the relevant dimensions of QoE studies?	Presentation and discussion of six dimensions transferred from other QoE domains
3.3.2	Is the validity and representativeness of study results affected by these dimensions?	Evaluation of quality ratings collected via user studies and varying design dimensions, i.e., interface design and domain knowledge
3.4	Is it applicable to integrate QoE studies into regular working processes?	Analysis of usage behavior and feedback of employees on a newly developed survey tool

Figure 3.1: Overview about addressed research questions and used methodology.

itoring QoE of business applications are discussed, especially with respect to their applicability in enterprises. Section 3.2 proposes the concept for monitoring QoE in enterprise environments. Section 3.3 focuses on the most important part of the concept – the evaluation of influence factors. In this context, relevant dimensions of the design of QoE studies for evaluating business applications are presented. The influence of two selected dimensions – the artificiality of the test system and the domain knowledge of study participants – on the QoE is demonstrated with two studies, mainly based on [4, 5]. Section 3.4 presents a survey tool which considers various requirements for conducting QoE studies with employees in enterprises, and its applicability. The section is based on [6, 7]. Finally, Section 3.5 concludes the chapter.

3.1 Background and Related Work

This section gives a brief introduction to the concept of QoE. Further, related works in the context of monitoring QoE of business applications is presented.

Focusing on the process of collecting quality assessments from the users, which is the most relevant part of the monitoring process, the importance of carefully designing subjective studies is highlighted and the influence of the design on study results is discussed.

3.1.1 Introduction to Quality of Experience

Today, application and service providers focus more and more on the needs and satisfaction of their users. This leads to an increasing importance of measuring and monitoring the performance of the services. However, as the concept of QoS does not consider the user's perception, meaning that a good QoS does not necessarily result in a good perceived quality on the user's side, the concept of QoE has been proposed by the network research community. QoE is defined as "[...] the degree of delight or annoyance of a person whose experiencing involves an application, service, or system. It results from the person's evaluation of the fulfillment of his or her expectations and needs with respect to the utility and/or enjoyment in the light of the person's context, personality and current state" [86].

Hence, the QoE describes the perceived quality of an application, service, or system by a user considering multiple factors in dependence on the system, the human nature, and the context of the usage. Transferred to the domain of business applications, when assessing the quality of the service in terms of QoE, the human factors such as the emotional state and expectations, technical aspects as well as the situation, e.g., using the software while communicating with customers, should be considered.

Due to the interrelation of such influence factors, classifying them in the three mentioned categories – human, system, and context – is not trivial [87]. The categories are multi-layered leading to a complex interplay of various facets influencing the user's QoE. Nevertheless, in the following, the most important layers of each category are introduced based on the categorization presented in [87].

Human influence factors comprise aspects such as age, gender as well as the physical, emotional, and mental constitution of a person. Further, personal factors including mood, personal values, attention, and motivation while using an application are part of this category. Besides these aspects, the QoE may be also affected by skills, expectations, and the user's background knowledge. For example, people who are experts rate the quality of a system more critical than people who are unfamiliar with a system [88].

The system influence factors describe technical parameters of an application, service, or system. These are related to all technical components such as the used devices on the client and server side, the network, but also content-related aspects. In this context, relevant network characteristics are, for example, bandwidth limitations, delay, and packet loss. Such network degradation may lead to waiting times for the users [89] and thus, affect their perceived quality of the application or service. An example for content-related aspects and configurations are the encoding and frame rate in the case of video conferencing applications which might result in differences in the quality perception.

The third category – the context-related category – includes physical- and activity-related circumstances. Physical factors are location-related and include, for example, place (inside/outside), temperature and light conditions, while activity-related aspects are focused on the activity of the consuming person, e.g., sitting or moving. Besides these factors, the social and economic background of the user is also relevant. In the context of QoE of business applications considering temporal- and task-related factors are indispensable. Temporal factors do not only include the daytime of usage, but also the frequency and the season. Seasonal effects may appear in business fields, e.g., in the health care sector, where quarterly reports are usual leading to a higher workload and more stress. This may alter the perception of the application performance.

One of the main disciplines of QoE monitoring is the evaluation of such influence factors to understand the relationship between the perceived quality on the user side and the measured technical performance of applications, services, and systems. This understanding is used to build models or define algorithms

to estimate the user's QoE based on technical parameters. Deploying the models in the application or network management, includes the user's perspective in these processes. This allows the allocation of resources in an efficient and accurate manner accompanied with satisfied users due to a good QoE.

3.1.2 Quality Assessment of Business Applications

For investigating the relation between perceived and measured performance of an application, the application's quality from the user's point of view needs to be assessed. This assessment can be done by using two commonly used methods, i.e., perception-based and instrumental approaches [86]. Perception-based methods describe the conduction of user studies where humans judge the quality of a single or multiple test conditions. To do so, a test environment is set up including specified test conditions. After introducing and training the participants, they run through the test and rate the quality under the certain condition. For the rating, often a five-point absolute category rating (ACR) scale is used [90]. By statistically analyzing the given ratings, an overall quality score for each condition is computed, e.g., the Mean Opinion Score (MOS) which describes the average score of the test participants [91]. In contrast, the instrumental methods compute the QoE, e.g., the MOS score, based on models or algorithms solely using input achieved from technical systems [86].

This chapter focuses on perception-based methods. Previous research mainly conducted such subjective tests in dedicated labs or online by using crowdsourcing. For the test design and conduction various recommendations published by the ITU Telecommunication Standarization Sector (ITU-T)[1] are available. The standards comprise recommendations for, e.g., quality of speech [92], multimedia applications [93], and web browsing [94]. As business applications and web browsing have many characteristics in common, e.g., sequences of interactions within sessions [89], the ITU-T P.1501 standard for web browsing notes that it might be also applicable for online business applications. However, applying the

[1] https://www.itu.int/en/ITU-T/; Accessed: August 1st, 2020

standard to user studies conducted within the enterprise is not possible without adaptations. Several challenges and requirements arise due to the enterprise context which are not considered in the recommendation, e.g., security and privacy considerations. These challenges are discussed in detail in Section 3.4.1.

In the context of business applications, first impressions of perceived quality has been derived from existing resources in enterprises without fielding dedicated user studies. As the IT sector in enterprises often provides support channels for their users, the perceived system quality was assessed based on the feedback and information given by employees via, e.g., ticketing systems [95] or system reports and requests for assistance [96]. However, these methods only provide very coarse-grained data and the evaluation is often difficult due to the unstructured information in the support requests. Approaches resulting in finer grained data usually involve active user feedback during or immediately after the use of the service or application. Examples for studies using this approach are Schlosser et al. [97] and Casas et al. [98]. Both works aim at a better understanding of the influence of varying technical parameters on the QoE of enterprise and related tasks like typing on a thin client. However, the tests were conducted in dedicated labs with students and not in a working environment with employees. Studies involving employees would overcome this limitation. Drawbacks of such an approach are, for example, additional costs on the enterprise's side and additional workload for the participating employees. A less cost intense approach, neglecting contextual factors, is collecting assessments from employees in online studies with a fictive business application [99] or in an enterprise lab [100]. In contrast, Smith et al. [101] introduced a method to collect feedback directly from employees in a real business environment. The feedback about the performance of the meeting software Microsoft Lync is collected at the end of each session via a survey realized as a game. While this approach is feasible for a software that is only used from time to time, it cannot be applied for applications that are used throughout the whole workday, like SAP[2] systems, as it would be too time and cost intensive to ask the users to rate the

[2] https://www.sap.com/products.html; Accessed: August 1st, 2020

performance after each interaction. Reducing costs by collecting ratings only once a day, e.g., after the last system interaction of a working day, would be also not feasible since the last interaction is not known in advance.

To overcome these drawbacks, this work introduces not only a concept for monitoring QoE in enterprise environments, it also proposes a survey tool to collect context sensitive quality assessments from employees while considering the trade-off between enterprise specific requirements and the granularity of the quality assessments.

3.1.3 Influence of Study Design on QoE

While designing subjective tests various aspects need to be considered as they may influence the reliability, validity, and representativeness of study results [102]. Therefore, one should be aware that not only test conditions play a relevant role when conducting experiments, also the test environment including technical and contextual factors, and the representative characteristics of the participants have an impact on the test results.

Regards the test environment, labs provide controlled but less realistic environments, whereas running tests online, e.g., with crowdsourcing, allows to realize contextualized subjective experiments. In this context, not only social and emotional aspects have a substantial influence on the QoE evaluation [103], it has been also shown that the results obtained using standardized experiments significantly differ from real-life QoE assessment in the case of video streaming [104, 105]. In contrast, QoE ratings for web browsing in real-world and employed laboratory tests are not affected by contextual factors or distractions [106].

Many crowdsourcing and laboratory studies focus on specific stimuli and try to keep the number of potential influence factors as low as possible. Consequently, the user's experience is often decoupled from real services, which means the test subject is not embedded into a realistic service environment, e.g., like YouTube or Netflix in case of video streaming. Thus, the question arises if

and to which extent the study design, and the interface design in particular, influences the QoE. Regarding web applications, studies revealed that the usability of web pages has a positive effect on the quality assessments [107, 108]. However, it is unclear to which extent the interactivity of the test interface influence participants' perception. Therefore, this work investigates this effect for a streaming service. It is evaluated if a realistic environment distracts the participants' attention such that the testtakers overlook service impairments and thus, if such an interactive interface changes the user's streaming experience.

Another essential aspect is the representativeness of test groups. Involving user groups with different background and characteristics, may affect the study outcome [109]. One reason for such an effect may be deviations in the users' expectations concerning the test subject. Previous research showed that, for example, the perception of experts deviate from those of non-experts. The direction of deviation depends on the investigated use case and its characteristics. Higher degradation results in more strict ratings from the experts in the context of video quality [88, 110]. This may be caused by different views on the test subject due to their domain knowledge in addition to higher expectations. In contrast, more complex test cases lead to the inverse effect due to the unfamiliarity of participants [111].

So far, studies only focus on differences between the perception of experts and non-experts, neglecting various nuances in between. For example, transferred to quality assessments of business applications, the perception and expectations of employees may differ from, e.g., people with crowdsourcing background as well as IT experts. Often, employees have been already working for years with the application and compare the test behavior to the known application behavior and situations during their work. Therefore, this work gives first insights into the perception of people with different domain knowledge and background by running a user study with experts, affected people and non-experts with and without microtasking background.

3.2 Concept for Monitoring and Modeling QoE in Enterprise Environments

Monitoring QoE of business applications raises new challenges on the technical side but also from the employees' point of view. From the technical perspective, the IT infrastructure is often highly complex dealing with multiple applications and system components which are administrated by different branches and persons. This makes it hard to identify affected applications as well as the most important technical parameters [112]. Further, companies might be reluctant to integrate monitoring solutions or influence a production system to solve these challenges. From the users' perspective, assessing the QoE of business applications and systems might only be meaningful in the work context, thus, involving employees in studies during working might be necessary. This is time consuming and potentially distracting, both negatively affecting the performance of the workforce.

Hence, a monitoring concept which considers these challenges is introduced in this section. The complexity of the monitoring process is reduced by an abstraction into three layers. These comprise the identification of performance issues in the enterprise application landscape, the identification of performance relevant parameters of the affected systems as well as of factors influencing the perception, and the estimation of the employees' satisfaction with the affected applications by building QoE models. Figure 3.2 shows a schematic overview of the proposed concept.

3.2.1 Identification of Performance Issues

The first step when monitoring QoE in enterprises is to identify applications and system components which are mostly affected by performance issues from the employees' point of view. This is not trivial, as employees have to work with a diverse set of applications and systems every day. Due to the often decentralized infrastructures, e.g., thin-client architectures, monitoring solutions have only a

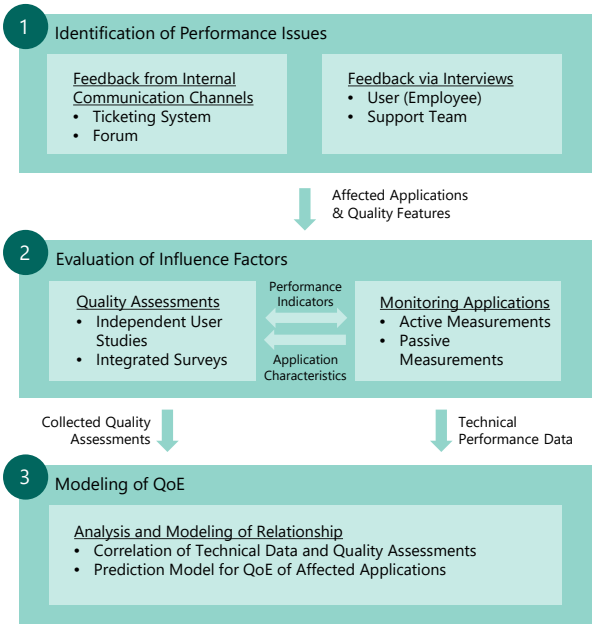


Figure 3.2: Concept for monitoring QoE in enterprise environments.

limited, not centralized view on the performance [112]. Regardless of this fact, some affected applications might dominate the perception of the users, making it indispensable to involve their perception and opinion. Besides the identification of affected components, performance indicators related to the perceived quality need to be extracted. Such indicators are, for example, loading delays or low speech quality of the telephone system. Knowing these metrics is essential to later on identify the related technical performance parameters for the further analysis of the relationship between QoS and QoE.

The identification of performance issues and related technical parameters can be done via direct or indirect feedback provided by the employees. By inter-

viewing the employees, direct feedback about the system performance can be collected. This leads to a trade-off between the precision of the feedback and cost factors. Conducting interviews results in a huge additional workload for the employees and the interviewers. A less cost intensive approach is deriving feedback from existing sources in the enterprise, e.g., forums, support channels, and ticketing systems. By filtering these sources for relevant information, performance-related content can be extracted and affected applications within the application landscape can be identified. The applicability of such a method was demonstrated by Zinner et al. [95]. Support tickets were classified with common text mining techniques. From these results most frequently affected applications as well as their quality indicators, e.g., loading delays on the client side, were derived.

3.2.2 Evaluation of Influence Factors

Gaining knowledge about the interplay between technical parameters and the perceived quality is import to understand the application's QoE. The acquisition of the knowledge is based on two parts, namely the identification of technical performance parameters related to the quality indicators and the evaluation of factors influencing the perception of the users. Analyzing the application behavior and evaluating influence factors are iterative steps. The detection of undiscovered influence factors require the identification of related performance parameters. Further, effects of changes in the system behavior on the user's perception need to be evaluated in user studies.

Before going into detail about the evaluation of influence factors, approaches for analyzing the applications behavior and identifying relevant parameters are briefly described. Based on measurement data, ideally containing performance data of the application and all used technical components, effects on the related performance indicators can be evaluated. Approaches for collecting performance data are active or passive measurements. Active measurements describe actively initialized measurements to analyze the system behavior under certain

(controlled) conditions. In contrast, passive measurements monitor the application behavior during the regular usage by the users. Thus, data from passive measurements, e.g., provided by existing monitoring solutions in the enterprise, gives insights into realistic system behavior [102]. A benefit from the passive approach is the possibility to derive meaningful ranges for relevant parameters, e.g., network delay or packet loss, later on used in the evaluation of the perceived quality on the user's side.

The evaluation of influence factors requires the assessment of the performance quality from the employees. The collection of such quality assessments requires the conduction of subjective experiments. These user studies can be realized independently or integrated into regular working processes. Regarding an independent approach, by building mock-ups from the business software interfaces for the affected processes or using sandbox systems, tests are decoupled from the employee's day-to-day work as well as from the production system. In this context, a mock-up describes a kind of fake software. The provided user interfaces look, feel, and behave like the original software without the software functionality in the background. For the test setup, conditions and a meaningful parameter space are derived from the previous analysis of the application behavior. Running user studies independently from working processes has several advantages. To name a few, there is no effect on business processes and the production system and the test conditions are controlled. Further, a conduction outside of the enterprise is conceivable. However, it has to kept in mind that external participants often have no domain knowledge. A major drawback of the independence of this study approach is the missing context of usage.

The second, integrated study approach overcomes this disadvantage. Such an integration could be realized by conducting surveys while the employees interact with the affected applications during their day-to-day work. However, this approach suffers from uncontrolled conditions which may lead to unwanted side effects, e.g., ratings may be affected by the bad performance of other system components. Furthermore, assessing the performance quality imposes distraction on the employees side as the employees need to focus on their reg-

ular work, especially when using the traditional pull approach for the collection of the ratings. The pull approach describes the method, where the test software polls ratings from the study participants. To reduce the distraction, a self-motivated approach is conceivable. Comparable to a complaint system, such an approach allows the employees to rate the performance at any time initialized by them selves. Besides uncontrolled conditions and the aspect of distraction, the integrated study solution raises additional challenges and leads to enterprise-specific requirements – these are discussed in Section 3.4.1 in detail.

3.2.3 Modeling of QoE of Affected Applications

The aim of monitoring QoE for business applications is not only to characterize influence factors, but also to estimate the QoE based on objective data. This requires an underlying QoE model. With such a model the QoE or the satisfaction of employees with the system performance can be predicted by using solely technical data, especially QoS data. The complexity of estimation models varies depending on the considered influence factors. Building generic models for all employees using affected applications might be inaccurate, as used processes and used functionality often differ due to a rich spectrum of tasks within a company. Hence, models specified on certain fields of tasks as well as personal models might be even more promising [8].

Nevertheless, while analyzing the relationship between subjective ratings and technical parameters the validity and the representativeness of the data should be taken into account. Regarding the validity, data collected without relatedness to the work context may not reflect the perception of the employees. On the contrary, context-related monitoring in the wild may suffer from uncontrolled conditions. In combination with data aggregation, commonly done in monitoring systems due to the huge amount of daily measured data, it may lead to noisy data sets. Furthermore, noise in subjective ratings due to side effects may also affect the validity of the results of the analysis. For example, such effects may be caused by seasonal events or temporal changes in the system,

e.g., software updates resulting in changes of the application behavior or affecting the user behavior due to changes in business processes. After changing processes or interfaces, users have to get familiar with them, especially in the context of business software [113]. Regarding the representativeness, subjective ratings and measured performance data need to be assessed with respect to their representative character. Due to voluntary participation the test groups may not represent the population, especially as the motivation to participate may change over time [114].

Modeling of QoE of business applications as well as the analysis of validity and representativeness of subjective and technical data is part of Chapter 4.

3.3 Dimensions of Monitoring QoE of Business Applications

While designing user studies for collecting quality assessments of business applications one needs to consider various dimensions affecting the validity and representativeness of the study results, outlined in Section 3.1.3. To get a better understanding of such effects, this section discusses dimensions related to the study environment and setup as well as related to the participants' characteristics. The influence on the validity and representativeness of the monitored QoE is demonstrated for both – setup-related and participant-related – categories of design dimensions.

3.3.1 Description of Dimensions

Designing subjective experiments for the evaluation of business applications comprises multiple dimensions. These dimensions are related to various facets of the study, e.g., the used methodology and tested conditions. The following discussion focuses on dimensions which might affect the validity and representativeness of the study outcomes. Figure 3.3 gives an overview about the included dimensions and demonstrates the influence on the results' uncertainty

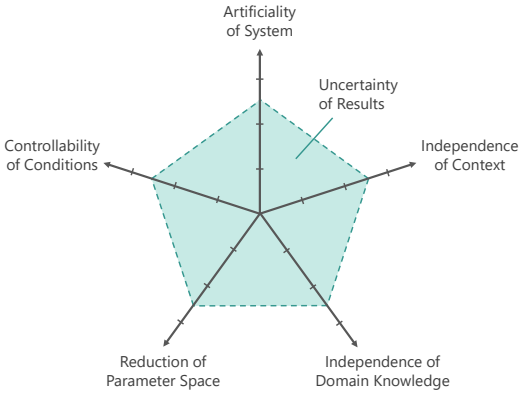


Figure 3.3: Dimensions of monitoring QoE of business applications and their relation to the uncertainty of study results.

of representing the perception of the employees with a spider plot. Positions on the axes that are further away from the center result in a more abstract study design and a more uncertain perspective on the QoE.

Starting with the dimension of domain knowledge, this dimension becomes even more relevant when evaluation influence factors on the perception of the users of business applications. Working with such applications requires often domain knowledge and thus, the perception on the quality might differ between experts and non-experts (see Section 3.1.3).

Study-related dimensions comprise the artificiality of the test system, controllable conditions, contextual factors and the size of the parameter space. These dimensions depend partly on each other. The artificiality of the test system describes how realistic the test system is. This includes not only the test software, but also the infrastructure used to run the test. An example for a realistic system would be running a study in the production system with the original software, whereas an artificial system would be, for example, a mock-up of the most im-

portant interfaces emulating software or network impairments, e.g., input or loading delays. The decision of the artificiality of the system influences the controllability of test conditions as well as the contextual aspects of the tests. Conditions of a study based on the production software are highly uncontrollable but context sensitive. In contrast, using a mock-up based approach leads to an entirely controllable situation, which, however, might end up contextless. Additionally, studies with artificial test systems conducted externally, e.g., via microtasking platforms, decrease the controllability of environment and conditions and further reduce the context-relatedness. While the dimensions of artificiality and controllability relate to the study environment, the size of the parameter space relates to the design of the study content. A more realistic behavior of the test subject might be realized by increasing the parameter space. On the one hand, this might increase the certainty of the study results, on the other hand, the higher complexity of the design might lead to unwanted side effects and inexplicable results.

To demonstrate and discuss the influence of dimensions of both categories, participant-related and study-related, on the monitored QoE, two user studies are presented. First, the dimension of system artificiality is investigated on the use case of video streaming. Second, the impact of the domain knowledge is demonstrated on an interdisciplinary, medical use case. Here, not only experts and non-experts are included, but the study considers also nuances in between.

3.3.2 Influence of System Artificiality

Many crowdsourcing and laboratory studies that investigate QoE focus on specific stimuli and try to keep the number of potential influence factors as low as possible. Consequently, the service experience in the QoE study is often decoupled from real services. This means, for example for video streaming, that the video under test is not embedded into a realistic streaming service environment like YouTube or Netflix. This widely used, artificial study design can be called *in vitro*, which means literally, “in a glass” or “in a test tube”. In contrast to this,

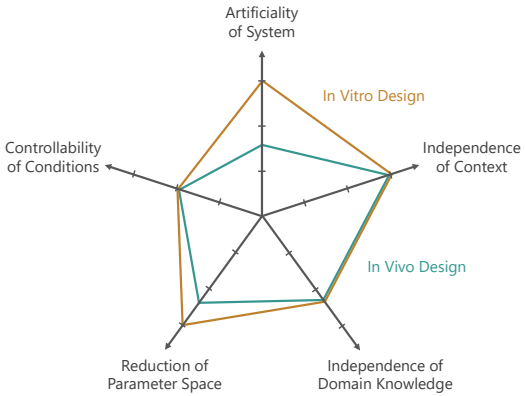


Figure 3.4: Dimensions of in vivo and in vitro study design.

it is also possible to design studies *in vivo*, which means literally, "in the living". Here, the study is conducted in conditions that precisely mirror those existing in real life, e.g., video streaming embedded in a real service like YouTube. With both study designs – in vivo and in vitro – and also multiple nuances in between in place, the question arises if and to which extent the study design, especially the interface design and its degree of interactivity, influences the test participants' QoE and attention to the shown stimulus.

Study Description

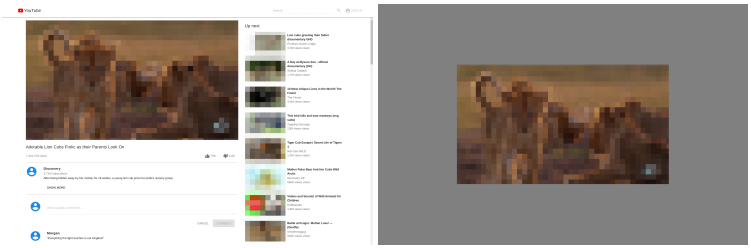
To answer this question exemplary on the use case of video streaming, a user study is conducted with an in vivo and in vitro interface design.

The design dimensions of both study versions are depicted in Figure 3.4. Both studies were realized via microtasking, neglecting the contextual factors and the domain knowledge of the study participants. Even if the crowdsourcing approach adds uncontrolled conditions, by considering best practices for quality assurance in crowdsourcing [44] as well as best practices for QoE crowdtest-

ing in particular [22, 115], these uncertainties are reduced. The variation of the interface design does not only affect the artificiality of the test system, it also slightly influences the parameter space.

The structure of both studies are the same as detailed in the following. After presenting an introduction to the subject of video streaming, the participants were asked to provide some demographic information, e.g., age and gender. In the main task, the participants had to watch one video that was shown either on a plain gray background (in vitro) similar to [116] or in a YouTube-like web page (in vivo). The video could be one out of three different videos, which covered a wide range of characteristics: A soccer match (fast motion), an animal documentary (slow motion), and a pop concert (motion, amateur recording). The videos were provided without audio to reduce additional factors influencing the perceived quality. Furthermore, the videos were displayed in high-quality (1080p), playing at least 25 frames per second, and had a length of 60 s. Each video playback was interrupted (stalled) twice – after 5 s and after 50 s of playtime. The stalling duration was 6 s each. The video was followed by questions about the content and if, how often and when stalling was noticed. Further, questions regarding the perceived annoyance of the stalling events and the perceived QoE of the streaming were asked. In a final questionnaire, the participants could provide some information about their usage of video streaming services. The full set of questions is in Appendix B.4. The answers given to all questionnaires were later checked for consistency and used to identify unreliable participants. Furthermore, the number of recognized stalling events were used to exclude cases with technical issues, e.g., more than two stalling events.

The video to be displayed as well as the design setting were randomly selected before a participant accessed a test. As the YouTube-like design includes the description field of the author who uploaded the video, comments of other users, and previews of suggested videos, the testtakers could scroll and could become distracted by other parts of the web page. Participants watching the video sequences on the gray background were not able to scroll. To analyze the users' interaction with the test software, the mouse position relative to the web page



(a) In vivo design – YouTube-like web page.

(b) In vitro design – Web page with video player on gray background.

Figure 3.5: Screenshots of the in vivo and in vitro study designs.

was tracked every 100 ms while watching the video. Also, the video position was tracked every 100 ms in the in vivo design. This monitoring technique, in conjunction with the screen size, allows to recognize if the page was scrolled and if the video was still in the visible range. Further, testtakers could interact with the in vivo web page by liking or disliking the video, displaying or hiding the whole video description, adding a new comment, canceling the addition of a new comment, clicking one of the suggested videos, and changing the window size as the design is responsive. After clicking on a suggested video, the participant was requested to answer the study questionnaire including a question to indicate the reason for clicking on the video. During video playback, all interactions were tracked for later analysis since these may lead testtakers to lose focus, i.e., the participants may not notice the stalling patterns due to interaction possibilities. An example of the realization of the in vivo and in vitro setting is shown in Figure 3.5.

Study Conduction

The studies were conducted on the microtasking platforms Microworkers and MTurk in November 2019. Independent of the platform, the participants received a reward of US$0.15 with an estimated time to complete the task of less than

Table 3.1: MOS values of ratings concerning the perceived video streaming quality including 95 % confidence intervals.

Video	Vivo	Vitro
Soccer	4.422 ($\pm$0.149)	4.422 ($\pm$0.171)
Animals	4.356 ($\pm$0.152)	4.293 ($\pm$0.164)
Concert	3.600 ($\pm$0.190)	3.704 ($\pm$0.196)

7 min. No further restrictions, like country or skill filters, were applied to limit the workers' access to the task. Overall, 822 participants completed the final questionnaire. Of those, 45.13 % did not pass the reliability checks resulting in 451 ratings for the evaluation.

Evaluation

To evaluate the influence of the interface design on the perceived streaming quality the MOS of the quality ratings is compared. Table 3.1 gives an overview about the MOS values including 95 % confidence intervals. Regarding the video content, only the MOS values for the in vivo and in vitro design are significantly lower for the concert than those for the other videos. A Kruskal-Wallis test revealed that there are differences between the video content ($\chi^2(2) = 77.809$, $p < 0.001$), and a pairwise comparison using the Wilcoxon rank sum test with Bonferroni correction establishes that the concert is rated significant lower than the other videos with $p < 0.001$. Comparing the ratings of both approaches, the MOS values and overlapping confidence intervals indicate that the ratings are not affected by the interface design. The observation is supported by the Mann-Whitney U test, resulting in no rejection of the null hypothesis with $p > 0.05$. Thus, no influence of the interface design on the perceived streaming quality could be found.

As the perception of stalling events may be affected by the design, the mean degree of annoyance is analyzed. Therefore, the mean degree of annoyance is compared, see Figure 3.6. As the number of stalling events affects the degree of

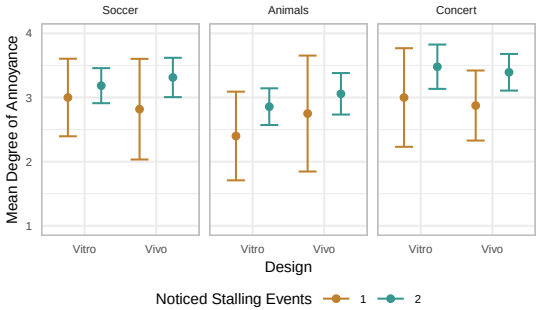


Figure 3.6: Mean degree of annoyance with 95 % confidence intervals.

annoyance [117], the analysis considers the number of noticed stalling events. The recognition of a single stalling event occurs less often (ranging from 9 to 16 participants per video), leading to large confidence intervals. Mann-Whitney U tests for one and two noticed stalling events result in no rejection of the null hypothesis that samples originate from populations with the same distribution. Again, no significant difference between the in vivo and the in vitro design is visible. Thus, the results show that the annoyance does not depend on the interface design.

Nevertheless, one can assume that the streaming experience is influenced by interacting with the web page. For the evaluation of this hypothesis only 230 participants are considered who watched a video in the in vivo design. Of those, 55.9 % interacted with the web page. The most often (98.4 %) observed type of interaction is scrolling the web page during the main task. Other page interactions were rarely used. Only 12.6 % of the participants interacted with the page in another way than scrolling. Regarding scrolling, 60.1 % of participants using the scroll functionality scrolled in a way that the video was sometimes no more visible on the screen (further referred as *scrolled out of focus*). The behavior concerning scrolling out of focus differs significant between the videos, indicated by a chi-squared test ($\chi^2(2) = 6.042$, $p < 0.05$). Scrolling out of focus is quite

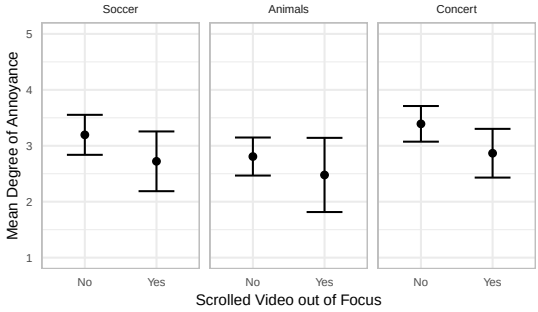


Figure 3.7: Mean degree of annoyance with 95 % confidence intervals for participants having the video always visible and those who scrolled it out of focus.

more likely when watching the concert (73.8 %) than for the soccer game (47.6 %). Thus, the video content has an impact on the interaction behavior of the users.

Regarding the perceived streaming quality when focusing on page interactions (excluding scrolling), participants who interact with the web page perceive a higher streaming quality with a MOS of 4.38. The MOS value for the group of participants who did not interact with the web page is about 4.07. A Mann-Whitney U test establishes that the samples do not originate from the same population ($W = 3\,540, p < 0.05$). Having a look at scrolling, no significant influence on the streaming QoE can be found.

Besides the influence on the perceived streaming QoE, an influence of interacting with the web page on the degree of annoyance of stalling is also expected. Nevertheless, no difference can be seen between the annoyance ratings of participants who did not interact with the website at all and interacting participants.

Focusing on scrolling out of focus, an effect is visible. This effect is visualized in Figure 3.7, which shows the mean degree of annoyance with 95 % confidence intervals for participants having the video always visible and those who scrolled out of focus. Here, the difference between ratings for the concert video is sig-

nificant, established by a Mann-Whitney U test ($W = 459.5, p < 0.05$). Participants who scrolled the concert video out of focus perceived stalling events as less annoying with an average rating of 2.87, while keeping the video visible results in larger annoyance with a mean degree of 3.39. A same, but not significant, trend is observed for the soccer game and the animals video.

Contrary to expectations, the study results show neither a significant influence of the interface design on the perceived quality nor on the degree of annoyance of stalling events. Nevertheless, they are in accordance to findings about the influence of advertisement banners described in [118]. On the contrary, the results indicate that the interactiveness of the study interface, mainly scrolling, has an influence on participants' focus. Participants start scrolling and thus, can lose focus on the stimulus. Even if this behavior is undesired during the evaluation of influence factors, one can argue that it is actually a more realistic behavior of users and thus, should be considered in the study design.

Finally, an influence on QoE caused by the way the user interacts with the study interface is observed. Interactions also tend to influence the degree of annoyance, if the participant scrolls the video out of focus. However, it is difficult to generalize these observations, as only few participants used the interaction possibilities in the current study. This might result from the fact that the participants were told to watch a video as part of a payed crowdsourcing task or due to apprehensiveness to impact test results by interacting with the page [119]. The test setting may lead to an unnatural behavior, in particular, a stronger focus on potential video impairments and less natural interactions with the web page. In a real life streaming environment, one would expect the users to interact more often, and in that case higher influences are to be expected since viewers might be less focused on the stimulus. In the context of business applications, the study results also encourage to test stimuli in a realistic environment that fosters natural behavior of the test participants.

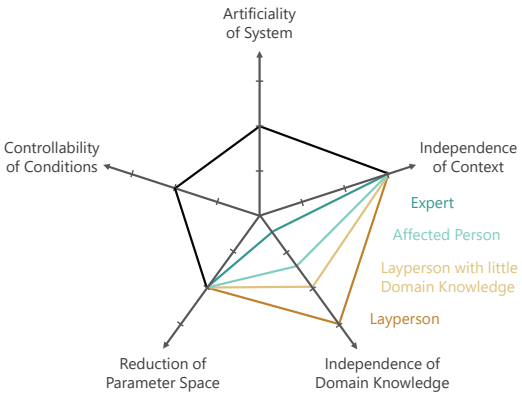


Figure 3.8: Dimensions of user study on varying domain knowledge and background of participating people.

3.3.3 Influence of Domain Knowledge

The influence of the domain knowledge and background on peoples' perception is investigated with a medical use case. Using a non-technical case for the study has been intentionally, as it is easier to classify people in groups with different background knowledge than for an interactive application. Figure 3.8 gives an overview about the dimensions of the user study involving four groups of participants with different nuances of domain knowledge, i.e., experts, affected people, persons with little domain knowledge, and laypersons. The group of laypersons contains participants with different background – microtasking and non-microtasking – leading to five groups of participants in total. For all groups, the other study dimensions are the same.

The main task of the user study is the assessment of deformational cranial asymmetries of newborns' heads. The head of a newborn is malleable, and therefore its shape is deformable, e.g., by resting on the same spot over a long time or due to prenatal reasons. Such deformations are called deformational cranial

asymmetries [120]. If the deformation is more advanced, a therapy for aesthetic and medical reasons is necessary. Even if there exist several objective metrics to classify [121] and to quantify the severity of the deformation, e.g., [122], these are not fixed thresholds when to start or to stop the medical treatment [123]. This is mainly because the subjective perception of the degree of deformation is not fully understood yet and may even differ between experts (physicians) and affected people, e.g., parents of newborns with deformational cranial asymmetries, as well as non-experts. Hence, the study subject is related to other QoE studies in the way that the people's individual perception of a stimulus and influence factors are evaluated.

Even if it seams far-fetched using crowdsourcing to solve medical tasks, there is a large community focusing on the usage of crowdsourcing in the context of medical research [124–126]. Some of the works already compare the perception of experts and workers, e.g., [127], showing that ratings from workers are reliable and well correlated to the assessments of the expert group. However, it is still unclear if the perception of people with different background differs concerning the severity of shown stimuli or subjects. Further, the authors did not consider the group of affected persons or family members, and other background information of the participants.

Study Description

The data set used in the study consists of 3D scans from 51 newborns' heads that exhibit different severity and types of deformational cranial asymmetries. The 3D scans were produced with the methodology described by Meyer-Marcotty et al. [128] that allows viewing the scans from different perspectives interactively. To reduce the number of samples, but still test a diverse and representative set, the data were clustered with the partitioning around medoids (PAM) algorithm based on biometric information. The clustering resulted in six clusters and thus, in six representatives for each cluster. These patients were as distinct as possible concerning their characteristics. Four additional patients were added based on suggestions of medical experts. This resulted in a data set of ten different pa-

tients to be assessed by the study participants. To further evaluate if an asymmetry on the left or right side of the head is perceived differently, mirrored versions of all scans were generated.

In order to guarantee that the scans could be viewed on all types of devices, the scans were converted to short videos of 30 s. Each video showed a rotating head omitting the frontal view of the newborn's face due to privacy policies. Below the video, the testtakers were asked if s/he (strongly) agrees or disagrees that the head is asymmetrical on a five-point absolute category rating (ACR) scale. In case the testtakers recognized a deformation, they were also asked to indicate the areas of the head where the asymmetry was noticed, e.g., at the back of the head or the forehead.

The study was realized as a web page and was structured as follows. After introducing the participants to the subject of the study, the videos of the scanned heads were shown to the testtaker, namely the original video of the scans of the ten selected patients and the mirrored version of these videos in random order. Two of the videos were shown twice to evaluate the consistency of the ratings. In total, each participant watched 22 videos in random order. Figure 3.9 shows a screenshot of the realization of the web page containing a video. After rating all videos, the participant was asked to provide additional demographic information, like age and gender. Further, background information was collected such as if the participant works in health care, if so, in which area as well as if the participant has small children, see Appendix B.5.

Study Conduction

The group of crowdworkers was recruited via the crowdsourcing platform Microworkers in March and April 2017. The study were limited to users from the United States, Canada, and the United Kingdom to prevent misunderstandings concerning the instructions due to language barriers. Further, the limitation reduces side effects due to demographic or cultural differences, e.g., aesthetic aspects. The payment per participation was US$0.50. The participants of the other groups, i.e., pediatricians, other physicians, parents, and other non-experts, were

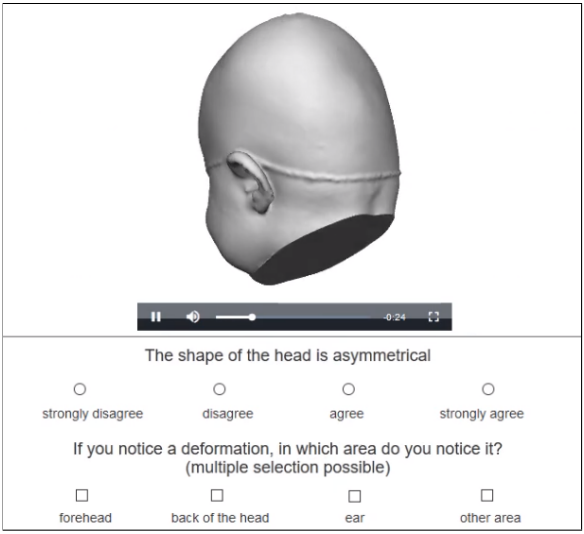


Figure 3.9: Screenshot of the web page showing a video of the rotating head.

invited via e-mail between July and September 2017. Here, participation was voluntary.

Table 3.2 presents an overview of the groups of participants. Only participants who answered all questions and gave consistent answers are considered. The groups of participants (slightly) differ in their demographics. The average age of the parents (38 years), other non-experts (42 years) and other physicians (41 years) is slightly higher than in the group of crowdworkers (32 years). Further, the group of pediatricians is the oldest (50 years) with the lowest share of female participants (25.1 %), while the group of parents has the highest share of female members (76.7 %).

Table 3.2: Overview about characteristics of the participating groups.

Group	N	Domain Knowledge	∅ Age	Share of Women
Pediatricians	31	High	50	25.1 %
Parents	73	High/Mid	38	76.7 %
Other Physicians	26	Mid	41	46.1 %
Crowdworkers	54	Low	32	46.3 %
Other Non-Experts	54	Low	42	61.1 %

Evaluation

To force the participants to form an opinion on the asymmetry of the shown heads, the neutral rating option was removed during the study fielding. This means, that the five-point rating scale was changed to a four-point ACR scale. To make the ratings still comparable, all neutral answers given on the five-point scale were excluded in the following evaluation. The remaining ratings, originally based on a five-point rating scale, were transformed to a four-point scale by reducing ratings of 4 and 5 by one. Hence, a rating of 1 represents the option strongly disagree and a rating of 4 means that the participant strongly agreed that a shown head is asymmetrical.

Before evaluating the impact of the domain knowledge on the perception, the consistency of the given ratings is analyzed.

Consistency of Ratings To evaluate the consistency of the assessments, the ratings of the videos which were shown twice within the study are compared. By using the Kruskal-Wallis rank sum test, it is analyzed if the ratings of the first and the second occurrence of the videos originate from the same distribution. As the test is not significant with $p > 0.05$, the null hypothesis – the two sample originate from the same distribution – cannot be rejected. This is an indicator for the consistency of the ratings. Nevertheless, Pearson's correlation coefficient of only 0.67 shows that some of the ratings are inconsistent. In the

following evaluations participants providing ratings of the videos shown twice, which are not identically or not located next to each other in the rating scale are excluded. Overall, five crowdworkers, two other non-experts, four parents, and one pediatrician are filtered out. Further, the assessments of the duplicated videos are omitted from the evaluation.

Comparison of Asymmetry Ratings To evaluate potential effects on the perceived asymmetry, the assessments of the participants per group are analyzed and the MOS is compared. Omitting the mirrored version of the videos from the following evaluation reduces potential biases caused by side effects.

By using the Kruskal-Wallis rank sum test, differences in the ratings between the groups are analyzed. The test shows a significant effect of the group on the ratings ($\chi^2(4) = 92.175$, $p < 0.001$). A pairwise comparison using the Wilcoxon rank sum test with Bonferroni correction revealed significant differences between the crowdworkers and all other groups ($p < 0.001$). By analyzing the ratings in detail, one can observe that 66.8 % of the crowd-based ratings (strongly) agreed that the shown heads are perceived as deformed. In comparison to the other groups with a percentage of agreements ranging from 35.8 % to 45.3 %, the amount is by far higher. Other than expected, this indicates that the crowdworkers perceive weak deformations as more critical than the other groups. Another explanation may be crowd-specific, additional influence factors on the assessments, e.g., the participants are less attentive due to distractions or rushing through the test. Alternatively, the phrasing of our question might induce bias, and the workers might assume that they are *expected* to identify an asymmetry.

As the observation may be invalid on a per patient basis, the assessments are also evaluated on a patient level. The average ratings per patient, including 95 % confidence intervals, are shown in Figure 3.10. While the average ratings for the groups of other non-experts, parents, pediatricians, and other physicians are quite similar with mostly overlapping confidence intervals, the crowdworkers more often agreed to notice a deformation. This leads to a higher average rating

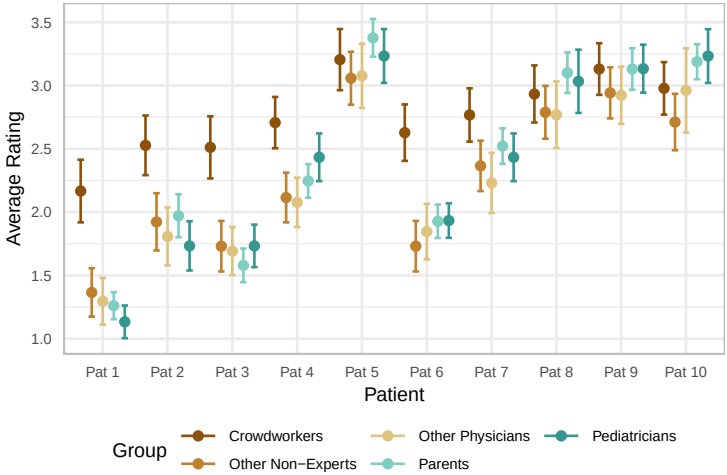


Figure 3.10: Mean of the ratings with 95 % confidence intervals per group.

for some of the patients. This observation corresponds to the result of a Kruskal-Wallis test per patient, which shows a significant difference ($p < 0.01$) between the ratings of the groups for all patients except Patient 5, 8, and 9. Differences are mostly observed for patients with less deformed heads from the perspective of study participants with no crowdsourcing background. For Patient 1, 2, 3, and 6, a significant difference between the assessments of the crowdworkers and those of all other groups is revealed by pairwise comparisons using Wilcoxon rank sum test with Bonferroni correction with $p < 0.001$. The ratings of the other groups do not differ significantly. Pairwise comparisons for Patient 4 also result in a significant effect between crowdworkers and the groups of parents, other physicians and other non-experts ($p < 0.001$). Differences between crowdworkers and pediatricians are not significant. Similar observations can be made for Patient 7. Here, only ratings from crowdworkers and other physicians as well as other non-experts deviate significantly with $p < 0.05$. Finally for Patient 10,

Table 3.3: Coefficients r of point-biserial correlation between ratings and areas of noticed deformation, i.e., front head, back of the head, ear and other areas, including level of significance.

Group	Front	Back	Ears	Other
Pediatricians	0.47^{***}	0.64^{***}	0.48^{***}	0.20^{***}
Parents	0.33^{***}	0.64^{***}	0.39^{***}	0.11^{**}
Other Physicians	0.27^{***}	0.57^{***}	0.38^{***}	0.19^{**}
Crowdworkers	0.09	0.42^{***}	0.30^{***}	0.18^{***}
Other Non-Experts	0.17^{***}	0.49^{***}	0.27^{***}	0.26^{***}

$^{**}\ p < 0.01$, $^{***}\ p < 0.001$

there is a significant difference between the group of other non-experts and parents ($p < 0.01$) as well as other non-experts and pediatricians ($p < 0.05$). These findings indicate, that for unique characteristics of deformation the perception differs between experts, including physicians and parents, and laypersons. This aspect is analyzed further by evaluating the assessments of the mirrored and original videos as well as the provided answers concerning the areas where the participants noticed the deformations.

Influence Factors on the Perceived Asymmetry As it may influence the perception if the deformation is located on a head's left or right side, the assessments of the original and the mirrored scans are evaluated. By using paired Wilcoxon's signed rank test, no significant differences between the ratings of the original and the mirrored videos for all groups could be found with $p > 0.05$. Thus, no impact on the perception is visible.

The relation between the ratings and the areas where the deformation has been noticed is analyzed, to get a better understanding of the testtakers' ratings. Table 3.3 summarizes the correlation coefficients per group between the ratings and the given answers. For all groups, a significant, positive correlation between the ratings and all areas except the forehead is observed. Regarding the forehead, the assessments from crowdworkers do not significantly correlate

with their answers, while for this option a significant, positive correlation for the other groups is seen. Correlations between noticed deformations at the forehead and the ratings are higher for the groups of pediatricians, other physicians and parents. This indicates, on the one hand, that for non-experts it is more challenging to identify deformations at the front of the head due to the missing view of the face of the newborns. On the other hand, it may be a piece of evidence that groups with different domain knowledge focus on different areas of the head, which may influence the perception.

To sum up, the results of the study showed that overall the perception of experts, affected people, and laypersons without microtasking background is quite similar. This observation is other than expected. Further, the perception of crowdworkers differs from those of the other non-expert group. Laypersons with crowdsourcing background more often perceived deformation of the shown heads. The analysis of the ratings per patient showed that some characteristics of deformations also lead to differences in the perception of the laypersons and the expert groups including the group of affected persons. The recognition of deformations is based on different areas of the head for experts and affected persons. Even if a frontal view of the faces is not shown, which makes it challenging to notice deformations on the forehead, they consider this area for their ratings. This observation may be an explanation of the different perception of deformational cranial asymmetries, as mentioned above. The differences concerning the focus of the participants do not fully explain the variations in the ratings, especially the differences of the crowdworkers and the other groups. Instead, these ratings may be influenced by crowd-specific, additional factors, e.g., inattentiveness due to distractions, biases induced by the phrasing of the instructions, or an insufficient training phase.

Overall, the results of the study gives evidence, that varying domain knowledge influences the perception and the focus of the participants. This should be considered when designing user studies, especially in the context of business applications. The study also shows the importance of carefully designing

subjective studies and experiments when conducting them with people without domain knowledge, especially with crowdsourcing background.

3.4 Designing a Survey Tool for Quality Assessments in the Wild

As microtasking-specific side effects may influence the results of subjective studies and the perception of affected people are more similar to experts than to laypersons (see Section 3.3), collecting quality assessments of business applications should be conducted with affected people, i.e., the employees. Further, integrating the assessment process into the regular working process contextualizes the ratings.

In this section, requirements for the realization of collecting ratings from employees continuously and at a large scale are discussed. Based on the presented requirements, a survey tool is introduced and its applicability is evaluated based on feedback obtained from two user studies in a large company.

3.4.1 Enterprise-specific Requirements

The following requirements are derived from discussions with a cooperating company and feedback of the participants of two pilot studies. Even if the ITU-T P.1501 standard for evaluating subjective studies for web applications notes that it might be also applicable to web-based business applications, the standard does not cover these requirements, especially for quality assessments in enterprise environments.

Minimization of Costs

While participants in lab or crowdsourcing studies solely focus on the assessment tasks, employees need to focus on their regular day-to-day work and the assessments impose additional work. Consequently, one requirement is the minimization of the effort for each participating employee. This makes a large num-

ber of questions impracticable and also the interface of a survey tool needs to be optimized to reduce the number of interactions per assessment. This can be realized by focusing on an interface with selectable items, e.g., check-boxes or radio buttons, instead of free text answers.

In addition to the assessment time, the number of participating employees needs to be minimized. Even if an employee can complete one assessment within a few seconds, scaling out the assessment process to all company employees can result in a significant amount of working hours per year. Thus, the number of participants needs to be dimensioned appropriately to generate representative results but also to limit required man power.

Seamless Integration

During the assessment of the perceived quality, the employees still have to complete their day-to-day work. This might include cognitive challenging tasks or personal contact to customers of the enterprise. Consequently, the survey tool needs to be seamlessly integrated into the existing workflows, or at least the imposed disturbance needs to be minimized. Further, unlike most other subjective test scenarios, the survey tool does not run on a dedicated test or evaluation system but needs to be integrated into the production system of the company. This imposes the need for additional security considerations and error handling, and also limits the available technologies.

The ratings of the users submitted through the tool have to be considered as privacy relevant data and have to be stored in a secure manner. Appropriate means have to be taken to anonymize the identity of the employees and strict regulations for accessing the data have to be added. Software errors potentially affect a large number of employees, thus, the main requirement for the error handling is that the software should fail gracefully. This means that in case of a software error, the software should be terminated without any notice of the user, even if this implies a loss of measurement data. Finally, most companies maintain a given software and infrastructure stack that defines the available technologies for the development of the survey tool.

Common Best Practices for User Studies

Despite the previous requirements and design limitations, the assessment methodology needs to be scientifically valid and respect established standards for subjective evaluations like ITU-T P.1501 [94]. Similar to other studies, the participants need some basic instructions regarding the survey process. Due to the large number of participants, a personal training is not possible and similar challenges to training phases in crowd-based subjective studies arise. Thus, the tool should be easy to handle and self-explaining. Further, the participation in the study should be not mandatory, thus, an appropriate motivation needs to be provided during the initial communication with the employee. The non-participation should not lead to disadvantages. Another important, motivation-related aspect is the prevention of implicit or explicit incentives for the employees to give a certain rating. Such an effect might occur, for example, if the required effort to complete a rating, e.g., the total number of clicks, is unequal for the rating options. If the options are followed by a different number of questions, it might incentivize participants to always select the shortest path for completing the study.

3.4.2 Tool Description

Based on the defined requirements, a simple tool for monitoring the QoE of employees is designed. The QoE monitoring is realized as a short survey which is shown via a pop-up to the participating employees. The survey comprises two steps shown in Figure 3.11. First, the user rates the system performance by clicking on a happy green colored or an unhappy red colored emoji. The green, smiley face represents a satisfying or good performance and the red emoji defines an unsatisfying or bad performance. A *neutral* option is intentionally omitted to correlate the ratings with technical measures and determine an acceptability threshold for those measures in future work. After rating the performance, the user may explain her rating by selecting one out of several predefined reasons. The application supports the functionality to customize these reasons and

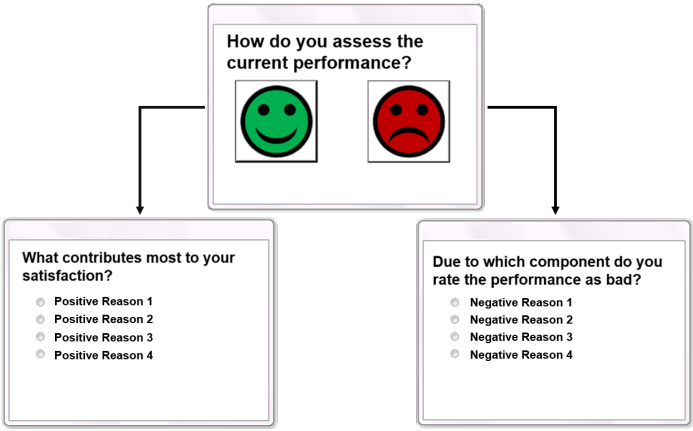


Figure 3.11: Graphical user interface.

create different subsets for specific groups of employees. To fit the needs of all participating groups, the possible reasons should be defined in cooperation with experts from the enterprise that already identified potential influence factors on the application's performance. Nevertheless, even if the set of reasons is optimized for the requirements of the specific group, not all possible performance issues can be taken into account. Thus, it is advisable to also add a reason *other*, in case none of the predefined reasons fits for the user.

Minimization of Assessment Time

To reduce the completion time of the survey, the number of questions is limited to a maximum of two. Further, the answer of each question requires only one click. After rating the performance by clicking on one of the emojis, the second question appears automatically. The interface of the survey tool is also automatically closed as soon as a reason is selected in the second step. Hence, the completion of the survey requires two clicks. Multiple choice is not allowed in

the second step, although this has been requested in the participants' feedback of a pilot study, as this requires an active submission of the survey resulting in a minimum effort of three clicks. For similar reasons the integration of an input field to enter an individual reason for the selected performance rating or other additional information is neglected. Text fields entail the risk that users enter sensitive customer data accidentally, as observed in a pilot study with the first prototype of the tool.

A further reduction of the number of required clicks and the completion time of the survey could be achieved by omitting the second step, i.e., the selection of a reason, for a positive rating. This might lead to implicit incentives to select the *faster* path through the survey Thus, the second survey step is mandatory for both – positive and negative – ratings.

Besides the number of required clicks, using colored icons instead of text buttons for the rating step further optimizes the assessment time. This reduces the amount of text in the pop-up and the colored emojis are easier and faster to identify.

Integration into Day-to-day Work

As discussed in Section 3.1.2, it is not possible to ask the employees to rate the system performance after each interaction with the system. Instead, the tool offers two approaches for collecting ratings, i.e., a self-motivated *push* method and a *pull* approach. The push system is similar to a complaint system. Employees are able to open the pop-up by themselves at any time, for example, by clicking on a desktop or a tray icon. Other than complaint systems, the performance of the system can be rated positively and negatively. In contrast, the pull system polls ratings from the employees by automatically opening the pop-up once an hour, if the user is logged into the system. Then the users are asked to rate the system performance within the last hour. This resembles the commonly used method for collecting assessments in QoE studies. In a preliminary version of the survey tool, the time difference between two pop-ups was set exactly to one hour. This resulted in the effect that the participants were *expecting* the pop-up

and prepared themselves to rate the performance. The ratings were not spontaneous anymore and in some cases working groups coordinated their responses. By varying the interval between two pop-ups, considering a minimal interval of 15 min and a maximum interval of 119 min, this effect is prevented. The alternative of binding the pop-up timer on additional thresholds, e.g., a minimum number of interactions of the user with the system, would require a tight connection of the survey tool with the production system of an enterprise. This is not applicable in each enterprise environment.

To prevent the interruption of critical working processes or conversations with customers, the pop-up is also closed automatically after a few seconds if the user does not react. These rating requests are marked as unanswered. The pop-up is also closed after a specific amount of time, if the user only rates the system performance and does not select a reason in the second step. In this case, the rating is stored and the reason is marked as missing.

Implementation

The tool comprises a client and a server component. The client side is written in C# and is automatically launched after logging into the system. As mentioned before, it is very important that software or configuration errors do not affect the employee in the daily work. Thus, the client component only supports a text-based error log but does not display any notification to the employee. Furthermore, before opening the pop-up, the client component sends a verification request to the server component including a predefined ID for the employee. The pop-up is then opened only upon confirmation of the server. This allows a remote administrator to easily stop the survey as a whole or for individual employees. After displaying the pop-up on the client side, the server calculates the next time the pop-up should be opened. The client software is put to sleep in order to save resources and again sends a pop-up verification request at the given point in time.

The server component is realized with a PHP framework. Besides the communication with the client component, it provides several options to configure

and manage surveys, e.g., the management of the predefined reasons and the groups of participants. The group management comprises manual creation of groups and automatic generation of samples of employees by a simple random sampling mechanism. Another purpose of the server component is the communication with the database to store the responses of the participants. Besides the ratings and the reasons, additional information is stored such as the time when the survey was opened at the client and the time stamp of submitting the response. These time stamps are required to determine the next rating time as well as to analyze the response behavior, e.g., concerning the time needed to complete a survey.

Recruiting Participants

The following communication concept is applied to inform the employees and to coordinate the study conduction. The concept considers the remote location of the participants and business settings by using e-mails as communication channel. A few days before starting the survey, the employees are invited via e-mail including information like the respective starting date of the survey and its duration. The employees are also informed about the purpose of the study and instructed how to use the tool. By going into detail about the goal of the study, the employees are motivated to participate. This observation was made based on feedback collected during a pilot study with the prototype of the survey tool. The participants realized that active participation can be used to improve their working conditions.

Regarding the usage of the tool, details about the design of the software is given, e.g., why a reason for positive ratings has to be selected. Additionally, the instruction highlights that the participants should rate the system performance within the last hour on a subjective and individual base, instead of coordinating their feedback. This additional information was also included upon request of the participants of the pilot study.

Beginning with the announced starting date, the survey is conducted which means the survey is shown once an hour to the participants during the specified

period. At the end of the survey period another e-mail is sent to the participants, thanking for their participation and asking for further feedback regarding the conduction as well as possible ways to optimize the tool. Further, a short summary of the results of the study is provided via e-mail to inform the participants and keep them motivated.

3.4.3 Analysis of Applicability

The applicability of the survey tool is evaluated based on two user studies conducted in cooperation with a company employing more than 15 000 people. The evaluation comprises the analysis of the integration into the regular working processes and the additional workload for participating employees, as well as the general acceptance on the employees' side. As the continuous collection of ratings with the pull approach might have a more negative influence on these aspects than the self-motivated push approach, the ratings were only collected with a pull system. Before discussing the studies' results, their setup and conduction are described briefly.

Fielded Studies

The two user studies *A* and *B* were conducted with employees working in different fields of business activity within the cooperating company. Due to the different tasks, the list of possible reasons for negative ratings used in the second survey step differs in study A and study B. These lists were created in cooperation with experts of the enterprise and included performance issues in a set of software modules required for the employees' day-to-day work and the option *other*. During two working weeks ratings were collected in study A from 618 employees in December 2015 and in study B from 723 employees in January 2016. After each study, feedback concerning the survey in general, its content, and the interface of the tool was collected within the company and passed on in aggregated form.

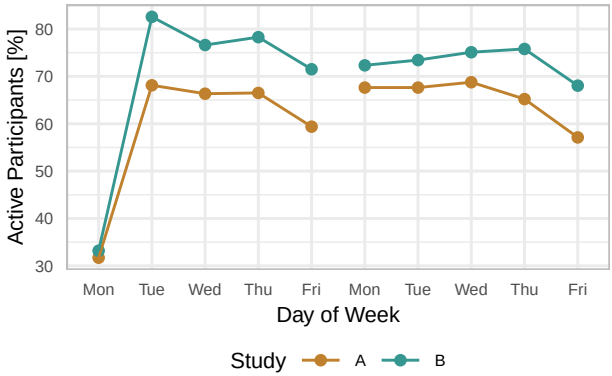


Figure 3.12: Share of participants submitting at least one survey per weekday.

Integration into Day-to-day Work

In total the survey was shown in study A 33 225 times with 16 339 rating requests marked as unanswered. In study B 47 113 rating requests were sent to the participants. Of those, 23 525 remained without an answer. This indicates that it is not always possible for the employees to answer the survey during the daily working processes. Reasons for this might be the completion of time critical work, talking with customers, or absence from their working place. Despite the percentage of missing answers, the results show about 97 % of the participants submit at least one survey during the total survey period of two working weeks. Figure 3.12 highlights the share of active participants per weekday compared with respect to the total number of participants invited in the study. A participant is classified as active, if s/he submits at least one rating within a day. Comparing the weekdays, the share of active participants ranges from 31 % to 69 % for study A and from 35 % to 85 % for study B. Possible reasons for the higher activity in study B may be the lower number of part-time employees in this user group as well as the fact that these employees mainly focus on data processing

and are less involved in customer care. Due to a ramp-up phase at the start of the study, the number of active participants on the first day is lower than on the other days. Indicated by the highest measured share of active participants during the two weeks, the ramp-up phase is finished on the second day. Comparing the remaining days, there are lower values measured at the end of each week. On Friday, the office hours in the company are usually shorter than during other working days. This, in conjunction with a number of participants working only part-time, leads to less employees participating in the survey. Except for the start phase of the study, the two weeks do not differ significant overall. Due to the continuous, active feedback collection once an hour the response rate per day is significantly higher than in other approaches with only a single response. Response rates less than 30 % are often observed when using such approaches for surveys conducted in enterprises [101].

Assessment Time

In order to evaluate the additional effort imposed to the employees, the overall response duration for study B is investigated. Missed rating requests and ratings without reasons are omitted. The median response time of the participants is 6 s. Due to some special cases with a response time of 132 s, the mean response time is significantly higher with 9.6 s. By using Wilcoxon's signed rank test it is checked if response times differ between positive and negative ratings. The input data is aligned as follows. Each employee e submitted m positive ratings $r^+_{e,1}, \ldots, r^+_{e,m}$ and n negative ratings $r^-_{e,1}, \ldots, r^-_{e,n}$. For the evaluation for each employee e the ratings R_e with

$$R_e = \{r^+_{e,1}, \ldots, r^+_{e,k}, r^-_{e,1}, \ldots, r^-_{e,k} | k = min(m, n)\} \tag{3.1}$$

and the corresponding rating times are considered. This results in total in $\sum_{e \in E} |R_e| = 6\,204$ considered rating times, with E being the set of all employees who submitted at least one rating during the study. Based on this subset,

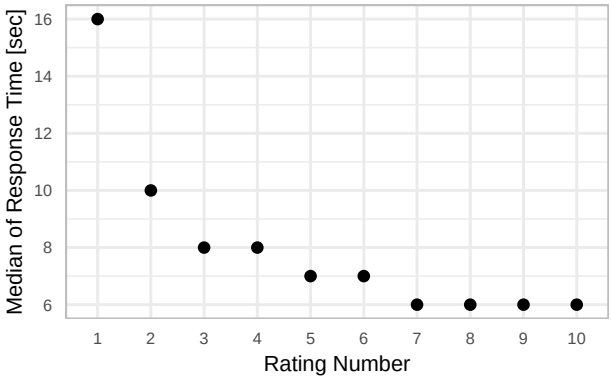


Figure 3.13: Time needed to rate the performance.

the difference between rating times for positive ratings and the rating times for negative ratings is not significant with $p > 0.05$.

To analyze if there is a speed-up in answering the pop-ups over time, the evaluation is limited to employees who submitted at least ten ratings. This applies to 79 % of all participants and their median answering time for the first ten ratings is shown in Figure 3.13. The rating times are measured at an accuracy of one second, as this is sufficient for an estimation of the additional effort imposed by the survey tool. The figure indicates that during the first seven ratings, the response times are decreasing from 16 s to 6 s, while the time remains roughly similar at 6 s after the seventh rating. Due to non-normally distributed response times and repeated ratings from the same employees Friedman's test is used to confirm the changing user behavior. The test clearly shows that response times for the first seven ratings significantly differ ($\chi^2(6) = 426.81, p < 0.001$), and that response times for the eighth to tenth rating are not significantly different with $p > 0.05$. Overall, one can assume that at the beginning of the survey new participants need to get used to the survey questions and the interface. In the

later course of the survey, participants can answer the questions more easily and efficiently.

The importance of a time efficient assessment process can easily be demonstrated by considering the total time t_t spend on the active submissions. The total time t_t can be calculated as the sum of all response times which results in $t_t \approx 38\,\mathrm{h}$. An estimation t_t' of the total time t_t based on the median response time of 6 s and the total number of answered rating requests $s_a = 23\,588$ in study B seams feasible, as $t_t' = 6\,\mathrm{s} \cdot s_a \approx 39\,\mathrm{h}$.

User Feedback

To gain insights into the employees' point of view concerning the conduction of subjective tests integrated into their day-to-day work as well as on the survey tool, the provided, aggregated feedback is presented and discussed in the following. The feedback is classified into three categories, i.e., general feedback, feedback regarding the survey content, and feedback related to the usage of the tool. The categorized, aggregated opinions are presented in Table 3.4.

The general feedback confirms the acceptance of the application by the users as discussed in Section 3.4.3. It shows that the users are willing to answer the survey and that they do not feel disturbed by the pop-up. Further, the feedback reveals psychological side effects. The participants suggest that there are no impairments of the used business application if the tool is running. Nevertheless, it is not possible that the survey tool influences the performance of the system.

The feedback concerning the study duration shows that the employees accept study periods of two working weeks. Asking the employees during a shorter time period is not suggested as it becomes even more difficult to observe additional influences on the performance due to temporal events, e.g., software updates, peak and off-peak times of the employees.

Regarding the survey content, some of the participants are irritated about the rating subject. They did not know if they should rate the performance of their last interaction with the system or if they should consider the system performance since the last rating. This irritation is not caused by the design of the

Table 3.4: User feedback concerning the survey tool.

Subject	Aggregated feedback
General	The employees are satisfied that the company is interested in their opinion. The survey receives in general neutral or positive acceptance. "Every time the pop-up has been shown, everything works fine."
Conduction	A study period of two weeks is appropriate. The provided information for the participants is sufficient and understandable.
Interface and content	The tool is easy to use. Sometimes it is not clear if the participants should rate the system performance between two pop-ups or the performance of their last system interaction. Some of the predefined reasons are ambiguous. The predefined reasons do not match the requirements of all working groups. The participants would like to provide additional information when choosing the reason *other*.

interface of the application as the users find it easy to use. Instead, it confirms the importance of providing sufficient and clear instructions to the participants.

Further, the participants gave feedback that the predefined reasons do not match the requirements of each participant. This is confirmed by analyzing the selected reasons for negative ratings. About 58.5 % for study A and 44.7 % for study B of the ratings were explained with the reason *other*. On one hand, it shows that it is difficult to fit the needs of all employees from different working groups with a limited number of reasons. On the other hand, it indicates that there may be additional factors which influences the perceived performance quality of the system, e.g., other system components that were not considered to be performance critical or usability aspects of the software. A possible solution is to gather those missing reasons directly from the participants via an

additional communication channel, e.g., e-mail or a discussion forum, during or at the end of the survey period. Due to the flexible implementation of the survey software, those reasons can easily be added to follow-up studies.

3.5 Lessons Learned

This chapter discussed the importance of monitoring the QoE of business applications in enterprise environments and highlighted the complexity of transferring existing monitoring standards and approaches to this domain. It presented a monitoring concept which includes the processes of identifying affected applications and components, evaluating influence factors on the QoE, and building a model to estimate the QoE which is valid and representative. Particular attention was paid to the design and conduction of user studies with employees as part of the evaluation process of influence factors on the QoE. In this context, the influence of the study design, i.e., the artificiality of the test interface, and the impact of the participants' domain knowledge on the perceived quality was investigated. Finally, a survey tool for collecting quality ratings from employees during their day-to-day work was proposed and its applicability evaluated with two large user studies conducted in a cooperating company.

The research questions, derived from gaps in the literature as outlined in Section 3.1, were answered as follows. The question about how to monitor and model the QoE of business applications in general was answered by proposing a monitoring concept in Section 3.2. By suggesting the usage of available resources in enterprises such as support channels, the minimization of time and cost factors are considered. The concept also pays attention to other enterprise specific requirements like the decoupling of subjective studies from the production system while considering the validity and representativeness of the resulting QoE model.

Section 3.3 addressed the question about important design dimension of user studies conducted in enterprises and their influence on study results. These dimensions are related to the test environment and content, i.e., the test system

artificiality, the controllability of conditions, and the size of the parameter space, as well as related to user characteristics, namely the domain knowledge, and the context of the applications' usage. The analysis of the impact of the artificiality of the test system on the quality perception indicated that there was only an effect between the interface design and the perception in case the study participants interacted with the test page. These participants lost their focus on the tested stimulus. However, the majority did not interact with the test interface. This behavior may be caused by the test situation. Regarding the influence of the domain knowledge on the QoE, the perception of people differed based on their domain knowledge, demonstrated by an assessment task of an interdisciplinary medical use case. The perception of affected people resembled the view of experts rather than the perception of laypersons. To sum up, the observations resulting from both studies gave evidence that the evaluation of influence factors on business applications should be done by conducting contextualized studies involving affected users, namely the employees.

The applicability of subjective experiments integrated into regular working processes and thus, collecting contextualized ratings from employees, was evaluated in Section 3.4. The introduced methodology is non-intrusive and offers the possibility to collect quality assessments continuously on a large scale. Studies with employees showed a high acceptance on the employees' side indicated by an outstanding participation rate with about 97 % of the invited employees submitting at least on rating during the study period. However, the feedback from the employees emphasized difficulties similar to challenges known from user studies on microtasking platforms, e.g., irritations about the rating subject. This highlighted the importance of providing good instructions when conducting studies unsupervised and decentralized.

4 Towards Estimating QoE of Enterprise Applications

Nowadays, employees spend a significant amount of their time interacting with enterprise applications as part of their daily work. Similar to other modern applications, business applications rely on distributed architectures, e.g., thin client computing, to benefit in terms of flexibility, scalability, and cost savings. In these architectures, functions such as storage and processing are performed in centralized data centers. Consequently, degradations with respect to QoS parameters such as network delays, packet loss, or load peaks in the data center can have a negative impact on the user-perceived application quality – the Quality of Experience (QoE). However, maintaining a high QoE is essential for achieving a high productivity of the employees and for avoiding frustration of the workforce. Hence, it is important to understand and quantify the impact of objective technical parameters like processing and transmission delays on the perceived quality of the employees during their day-to-day work.

Assessing the application quality is possible based on responses from application-specific feedback channels, e.g., ticketing systems, or based on quality ratings collected in user studies in enterprises. The latter approach provides a more precise view on the QoE. However, running such studies with employees in parallel to their regular work continuously and at a large scale is time and cost intense (cf. Section 3.4.3). Efforts can be reduced by estimating the QoE with a QoE models. These models are based on objective metrics, e.g., end-to-end network delays or total response times of applications. Nevertheless, quality assessments from the users are required for learning the model. Integrating

the trained model into the application monitoring or managing solution in the enterprise allows to assess the application quality from the users' perspective based on QoS parameters, identify performance issues, and initiate appropriate countermeasures in a timely manner.

Building a QoE model and implementing a system that is capable of reacting to QoS and QoE requires a reliable monitoring of technical parameters. Additionally, quality assessments from the employees need to be collected, e.g., during a user study of limited duration. Monitoring the QoE with different approaches, i.e., by actively asking the users to rate the quality (pull approach) or by using a self-motivated system (push approach), may result in differences in the provided view on the perceived quality. While the pull approach might result in a continuous, but less detailed view, the self-motivated push approach is more sensitive to short time performance issues, but may suffer from a decreasing motivation to provide ratings over time. Regarding the performance parameters, challenges arise from the technical monitoring deployed in a production system of enterprises. The data may be noisy due to measurement inaccuracies, limited access to all system components, or data aggregation to solve big data challenges, as discussed in Section 3.2.2.

These aspects give rise to several research questions. Regarding the collection methods of quality assessments, does the view on the QoE and conclusions on the users' satisfaction depend on the used method? Focusing on the QoE modeling, is it possible to model the QoE based on technical monitoring data which might be subject to measurement inaccuracies? How to quantify effects of real-world measurements, e.g., noisy data, on the model performance?

Figure 4.1 gives a brief overview about the research questions addressed in this chapter and the methodology used to answer them. Besides the evaluation of differences between the pull and push rating approach, this chapter focuses on analyzing correlations between the user ratings and the technical parameters. This analysis provides results for the data aggregated on different levels and investigates the impact of seasonal effects, timely changes in the system behavior, and effects caused by working on different tasks on the quality per-

Section	Research question	Methodology/Contribution
4.3	What are the differences in the view on the QoE of pull- or push-based approaches?	Analysis of usage und rating behavior of employees for both approaches
4.4.1	Are relations between pull-based ratings and technical parameters impacted by additional influence factors?	Correlation analysis of pull ratings and technical parameters w.r.t. different side effects
4.4.2	How to build a QoE model from pull-based ratings which can handle noisy and inaccurate monitoring data?	Development of a two-threshold-based model and quantification of unpredictable data
4.5.1	Are there relations between push-based ratings and technical parameters?	Correlation analysis of self-motivated push ratings and technical parameters
4.5.2	Is it applicable to model the QoE based on self-motivated ratings?	Development and evaluation of a machine learning-based QoE model

Figure 4.1: Overview about addressed research questions and used methodology.

ception. Finally, QoE estimation models are developed, evaluated and compared with respect to the impact of measurement inaccuracies and noise on their performance. The evaluation is based on QoE and QoS data collected from more than 4 000 employees in a long-term user study.

The remainder of this chapter is structured as follows. Section 4.1 gives an overview about factors influencing the QoE of interactive web- and network-based applications. Then, relationships between the main influence factor – waiting times on the user's side – and objective metrics are highlighted. Finally, models which leverage these relations are presented and related work is discussed. Section 4.2 describes the user study conducted during a study period of 1.5 years in a cooperating company and the resulting data set. On the basis of [9], Section 4.3 analyzes and compares the view provided on the QoE by pull and push approach. Section 4.4 sheds light on the correlation between

user-provided pull ratings and technical performance parameters under consideration of additional factors influencing the perceived quality. These factors are seasonal effects caused by vacation times, changes in the system behavior and user-related aspects such as working on different tasks and associated expectations. By leveraging the found correlations a threshold-based QoE estimation model for the pull approach is introduced and evaluated. The section is mainly based on [10]. In a similar manner, Section 4.5 analyzes the relationship between self-motivated push-based ratings and technical monitoring data. A machine learning-based model is developed, evaluated, and compared with the threshold-based approach, based on [9]. Finally, the chapter is concluded in Section 4.6.

4.1 Background and Related Work

This section gives an overview about factors influencing the QoE of web- and network-based applications and business applications in particular. Furthermore, associated performance parameters are discussed and approaches to predict the QoE based on such objective metrics are presented. In this context, not only related work on QoE models for business applications in general is reviewed, but also literature related to models build on self-motivated ratings similar to bug report or complaint systems is discussed.

4.1.1 Factors Influencing the QoE of Interactive Web Applications

Nowadays, business applications are often operated remotely in data centers. They are realized as three-tier architecture containing a presentation, application and data layer [129]. The integration of browser-based user interfaces simplifies the access and makes it more flexible. Thus, the characteristics of such applications might be similar to interactive web or even network-based applications, such as browsing. For both types of applications, user interactions

occur in sequences during sessions and protocols used for the communication might be the same, e.g., HTTP(S) [130], or are based on the same foundation, e.g., TCP/IP [131].

Until now, most QoE studies focus mainly on (web-based) applications and services for end-users, but enterprise applications are mostly neglected. In the context of web-based applications, the loading delay is the main influence factor on the perceived quality, highlighted in numerous studies, e.g., [89, 132, 133]. If a loading delay occurs, the user has to wait until the requested content is loaded resulting in a degradation of the perceived quality. It can also be shown that waiting times directly affect the user behavior, e.g., leading to a lower number of interactions [133–135].

Multiple circumstances lead to waiting times in web-based, interactive applications. On the network layer, network characteristics such as the bandwidth and its fluctuations influence the loading characteristic of an application [136]. Using mobile networks, handover processes, the strength of the signal [137], and the time to reestablish a connection after idle states of devices [138] lead to additional delays. On the application layer, used protocols, e.g., HTTP/1.1 or HTTP/2 [139, 140], as well as loading and rendering processes of the browser affect the time needed to load the content of a web site. Browsers and operating systems differ significantly in loading delays caused by these processes [141].

Indeed, the implementation of the application and its interfaces play a major role. In this context, the time until the content is visible and usable on the user's side depends on multiple factors, e.g., the type of content (text, images, multimedia content, CSS files), the elements' size, their request order, and the resulting order in which elements become visible [142]. Studies on the user behavior showed that users do not wait until the entire application or web page is loaded. They start to interact if areas or content which are relevant for their intended task are visible [143]. Thus, loading these areas and elements earlier than unnecessary content has a positive effect on the QoE [144]. In contrast, loading failures of elements such as images or CSS files impact the user's perception negatively [145]. This observation might be related to visual appeal and usabil-

ity aspects which are also influence factors on the QoE [107, 108, 146]. Users are more tolerant to loading delays if they perceive a web site as easy-to-use and aesthetic [107].

While many studies focus on influence factors from the technical perspective, there is little research on human influence factors in the context of QoE for web-applications. In [147] the authors observed that the expectation of users varied depending on the technology used for network access. Comparing wireless and wireline access, users rated the QoE of web browsing less strict while using wireless network access. Further, temporary network degradation influences the expectation of the users concerning the service quality and, thus, affect the QoE [148].

Regarding contextual factors, no clear effects of real-world distractions on web QoE could be observed [106]. However, there is a relation between the intended task of the user and her perception of the quality [132]. The QoE is influenced negatively by task characteristics such as a longer completion times and the overall length of sessions.

A first indication of the influence of delays in the context of business software is given by Bonhag et al. [99]. The authors investigated the perceived quality of a fictive business application by emulating loading delays for different types of tasks. The results show that the QoE is affected by a delayed application performance. The perception depends on the type of task and associated expectations. Users were more tolerant of delays of tasks which they knew that the application needed longer to complete, e.g., generating a report. Another study showed that working with delayed business applications affected the perceived complexity of tasks negatively [98]. The analysis focused on applications operated remotely on a thin client architecture. Delays in such applications are not only caused by network conditions, the configuration of the remote software also affects the response time [97].

Even if these findings give an impression how impairments like delays influence the perceived quality of users of interactive systems, it is not clear how the system performance and especially delays are perceived by employees us-

ing network-based applications in their day-to-day work. This work addresses this gap by analyzing the relationship between the perceived quality of a business application and different performance parameters. The analysis is based on data collected from employees while they were working on regular working processes in a long-term user study. Besides the analysis of correlations between quality ratings and performance measurements, additional influence factors are evaluated and discussed. The investigated factors are related to side effects from seasonal events, i.e., vacation times and system updates, and to expectations of the employees in dependence of their field of work.

4.1.2 QoE Modeling

Besides understanding the quality perception of users by investigating influence factors, service or network providers aim to integrate the user perspective into their application or network management. This allows the efficient management and usage of resources while considering the needs and perspectives of the users. In [149] the authors describe a concept for QoE aware traffic management. Here, application or network monitoring data is enriched with data obtained from QoE monitoring on the client's side. Based on both resources the providers make traffic management decisions to meet not only service level agreements (SLAs) but also provide network conditions that satisfy the users.

Collecting QoE monitoring data, i.e., quality assessments, continuously with user studies would be time consuming and cost intense. Hence, using a QoE model to estimate the perceived quality is more efficient. A QoE model approximates the QoE, e.g., MOS values, based on objective metrics related to influence factors, e.g., the measured waiting time. Regarding interactive web or network-based applications, there are multiple starting points to measure a diverse set of parameters, e.g., on the network or application layer as well as on the server or client side. As there is little research on QoE modeling for enterprise applications, again, the focus is on QoE models for web- and network-based applications. These web QoE models are often based on user-centric metrics associated

with the loading processes of web sites. Such metrics are, for example, the page load time (PLT), above-the-fold (ATF) metrics or Google's Speed Index. While the PLT measures the total time needed to load the complete content of a web site [89], ATF metrics define the loading time of the visible part of the content [150]. In contrast, the Speed Index[1] is related to the loading process itself. It describes how quickly is the visual progress while loading the above-the-fold part of a web site. For multiple of these loading process-related metrics a logarithmic relationship between the waiting time and the perceived quality by the users was found [151, 152]. Furthermore, numerous mathematical models exist using a single objective metric, e.g., Speed Index [153], page load time [154], or network bandwidth [151], combinations of objective metrics, e.g., mobile network factors such as handovers [155], or combinations of objective and subjective metrics, e.g., loading times and aesthetic aspects [156, 157].

Another, often used approach, is learning a QoE model based on multiple parameters with machine learning techniques. The parameters are taken from the network, e.g., flow characteristics [158], or from the application layer, e.g., variants of PLT and ATF [159]. Other models combine network parameters with user-related factors, e.g., age and education [160], or subjective factors such as perceived learnability and usability [161].

A first QoE model for a business application was introduced in [8]. The authors compared different machine learning approaches to predict the QoE on a binary scale. The evaluation showed that building a generic model based on measurements from an uncontrolled environment is difficult due to inaccuracies and noise in the data.

All of the mentioned models predict the QoE derived from quality assessments collected with the traditional pull approach. To recap, by using the pull approach study participants are asked actively by the test software to rate the perceived quality. Using this approach in the context of business applications and especially for collecting ratings from employees may result in coarse-

[1] www.sites.google.com/a/webpagetest.org/docs/using-webpagetest/metrics/speed-index; Accessed: August 1st, 2020

grained view on the QoE, as demonstrated by [8]. A different, more detailed view on the QoE may be achieved by collecting self-motivated ratings comparable to complaint systems, where users can report bugs and malfunctions of software or services. Learning a QoE model from such ratings may lead to a estimation model which is more sensitive towards short-term changes in the application performance. However, there is little knowledge about the correlation between self-motivated ratings and technical parameters. In this context, complaints may be related to QoE ratings in a certain way [162], even if it is difficult to map them to the often used five-point ACR scale [163]. Further, there is a relationship between user complaints and technical performance parameters as demonstrated for an IPTV service [162].

Following this self-motivated push approach to monitor the QoE of business applications, it is still unclear if these ratings are correlated with technical performance parameters. Furthermore, when building an estimation model on self-motivated ratings the model representativeness and validity might suffer from changes in the motivation to provide ratings over time. While other self-motivated feedback systems, e.g., product reviews, lead to a visible outcome to the feedback provider, reporting performance issues often has no direct benefit to the users. Thus, the motivation of the users differs depending on the perceived usefulness of the ratings. The analysis of user behavior with an integrated error reporting system by Microsoft, Inc. showed that the perceived usefulness is affected by the transparency of the data usage and transparency concerning the role of the users [164]. Further, the motivational factor may change over time [114]. Here, the authors investigated the activity and retention of volunteers participating in online studies. Motivational factors might not only affect the rating frequency, but also might lead to influences directly on the QoE. Such effects were shown in crowdsourcing-based user studies with paid participants and volunteers. Volunteers tended to provide lower ratings than the paid participants [165].

To close the highlighted gaps in the literature, both approaches – pull and push – are compared and analyzed concerning the usage behavior by employees

of a cooperating company and their motivation to provide ratings. From these analyses, conclusions are drawn on the resulting view on the QoE. Furthermore, a threshold-based model to estimate the QoE derived from pull-based ratings is developed. This model is able to handle uncertainties in real-world measurement data. Finally, the applicability of building QoE models based on self-motivated ratings is demonstrated by using also a threshold-based approach as well as techniques from the area of machine learning.

4.2 Long-term User Study

The analysis of different approaches to collect quality assessments and the estimation of the QoE of business applications are based on data collected in a long-term user study conducted in a cooperating company. The company operates in the business domain of health insurances. Thus, the tasks of the employees are diverse including customer services, financial processes, and human resource management. For processing these tasks the employees mainly use an enterprise resource planning (ERP) system which is based on the SAP ERP system[2]. An ERP system is a software system or suite of applications that integrates functionality for all fields of business processes within a company such as accounting, sales, customer services, and human resources [166]. From the technical perspective the ERP system is structured as three-tier architecture including a presentation, application, and data storage layer. The presentation layer is, for example, operated as thin-client architecture. In a thin-client architecture users access the application from a terminal via a remote virtual desktop while the application is running on central servers in the data center. Besides the ERP system, additional applications and systems, e.g., telephone and printing systems, are part of the daily working processes of the employees in the company. However, this work focuses on the performance of the main business application – the ERP system.

The application data used in the evaluation consists of ratings of the performance quality provided by the employees and of monitored technical parame-

[2] www.sap.com; Accessed: August 1st, 2020

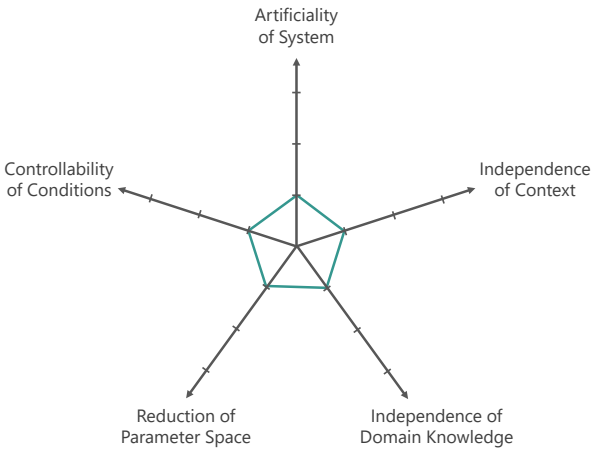


Figure 4.2: Design dimensions of a long-term user study to evaluate and model the QoE of a business application.

ters extracted from a monitoring system used in the company. The process of data collection of both subjective ratings and monitoring data is described in the following as well as the resulting data set.

4.2.1 Study Description

The user study was designed to run on a long-term and at large scale. Figure 4.2 shows the setup of the dimensions of the study design. To consider factors such as domain knowledge and the context of usage during the evaluation, the user study was integrated into the regular day-to-day work of the employees. Hence, all participants had a high level of domain knowledge and the performance was rated during the regular usage of the business application. Furthermore, the monitoring data was extracted from the production system. This resulted in a non-artificial test system. Monitoring in the wild leaded to highly uncontrolled

conditions as well as to a large parameter space. On the on hand, this study setup allows the evaluation of the QoE under real conditions leading to valid study results which represent the perspective of the employees during the regular usage of the application. This low uncertainty is illustrated in the figure by the area delimited by the green lines. On the other hand, the uncontrolled conditions may interfere the evaluation of influence factors on the users' perception.

Regarding the generalization of the findings and models presented in this work, answering the research questions by evaluating data of a specific application indeed leads to results which are specific for the investigated application, e.g., correlation coefficients and thresholds. Nevertheless, conclusions about the view on the QoE drawn from differences of the pull- and push-based systems are independent from the specific characteristics of the business application. Furthermore, lessons learned from the correlation analysis and consequences for the proposed modeling approaches are transferable to other applications and services used in enterprises.

Collection of Quality Assessments

The quality ratings were collected with the non-intrusive survey tool introduced in Section 3.4.2. Briefly summarized, the survey comprises two steps. First, the employees were asked to rate the performance on a binary scale. Second, they selected one reason for a good rating or the affected system component in case of performance issues out of a list of predefined options. The lists were provided by experts from the cooperating company and each included the option *other*.

The survey tool was configured in two ways to collect ratings with different collection approaches – pull and push – in parallel. The pull approach resembled the poll method commonly used in QoE user studies. Here, the tool actively asked the employees to rate the application performance once an hour by automatically opening the survey in a pop-up. The minimum distance between two rating requests was set to 15 min. As the employees worked permanently with the business application, it was not possible to ask for a rating after every interaction with the application. If the employees did not react within a short

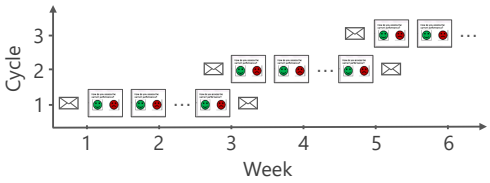


Figure 4.3: Schematic presentation of the study conduction with groups of participating employees changing every two weeks.

time span, the pop-ups were closed automatically, and these unanswered requests were saved as missed requests. The push system was realized by giving the employees the opportunity to open the survey by themselves at any time. To do so, a tray icon was integrated in the task bar of the operating system to make it as easy as possible to open the survey and provide ratings. Again, if no rating was submitted within a short time span, the survey window was closed automatically.

Due to the huge number of employees working with the business application, it was not applicable to let them all participate in the user study. The limitation of the study access to a few employees participating over years was also not feasible. The challenge was solved by running the study with changing groups of participants. To achieve representative results, the tool automatically selected user samples from the total population. For the calculation of the sample size a confidence level of 90 % and an accepted error of ± 5.5 % was used, both defined by the company. Each group was invited for a study period of two weeks. The members of each group were only able to participate in the study during the specified two weeks, further referred as a study cycle. Hence, every two weeks the group of participants changed. If an employee was not able or willing to participate in the study, the tool offered a sign out functionality. Thus, no further rating requests were sent to that user. Figure 4.3 depicts a schematic presentation of the study conduction exemplary for three study cycles. The x-axis represents

the time in weeks and the y-axis shows the groups of participants. In the following the process is described based on the transition of Cycle 1 to 2. Before starting Cycle 2, the selected employees were informed about their participation and instructions were provided. Details are presented in Section 3.4.2. During the invitation phase, only members of the current active cycle (Cycle 1) were able to rate the application's performance. At the end of Cycle 1, the survey was deactivated for members of Cycle 1 and activated for the second group. A short report about the results of the completed cycle was sent to the participating employees.

Monitoring Technical Parameters

The monitoring data was extracted from an external monitoring solution used in the company. While monitoring solutions integrated into the application provide the capability to collect detailed traces of application parameters and implemented operations [167], an external tool allows the investigation of the application in a holistic manner by taking network and system-specific performance factors into account. The monitoring data was measured with Dynatrace[3] Data Center Real User Monitoring (DC RUM). This tool uses a passive measurement approach by simple collecting network traffic from different points in the data center. The traffic is mirrored at traffic access points (TAPs) and switches and send to agentless monitoring devices (AMDs) for further processing. These devices extracts performance parameters related to the network, applications servers, and data base servers, e.g., throughput and response times, from the packets and flows [168]. The parameters are associated with a transaction triggered on the client side. Here, often a single application interaction results in multiple associated operations in the application, further called transactions. For example, loading a web page results in multiple XML calls and leads to multiple entries in the monitoring data. Due to the huge amount of data, the tool is not capable of exporting the data for every single transaction. Instead, the data

[3] www.dynatrace.com/; Accessed: August 1st, 2020

is aggregated by computing the mean of the values in intervals of five minutes per transaction type per user. This also includes transactions that run in the background or batch processes that are not necessarily triggered by the users. Hence, performance issues of these transactions are not necessarily recognizable by the users and thus, lead to noise in the data set. The performance of a QoE model may suffer from such noisy data.

The exported monitoring data comprises 22 performance metrics and 9 system related parameters. Here, values such as the total response time, the processing time on the server, the delay for traversing the network, and the number and size of the transferred packets, as well as flags for different types of errors are included. System related parameters give general information about the system, e.g., at which server the terminal client is running and which module or component of the business application is used by the transaction. For a table listing all parameters see Appendix C.

4.2.2 Data Set

The data set contains technical data and user ratings collected between mid-November 2017 and mid-August 2019. Data from regular holidays, e.g., Christmas, and days with incomplete technical monitoring data were removed from the data set. Further, for the evaluation only core working times from 6.30 until 18.30 were considered. Due to a misconfiguration of the monitoring tool during the first week of the analyzed time span, the technical data of this time span contains multiple entries of the same transactions. Removing all these duplicates was not possible due to the structure of the data. This additional noise can be neglected as it affects only a small part of the data and thus, has no significant impact on the evaluation. In total, the data set contains 70 651 831 entries of technical data.

Regarding the quality assessments, two study cycles were excluded due to technical issues with the survey tool. During the remaining 43 study cycles 4 433 employees participated in the study. Of those, 4 320 participants provided

389 105 ratings via the pull approach and 28 632 ratings were collected with the push approach from 3 207 users.

4.3 Analysis of User Behavior for Push and Pull Systems

Learning a model to estimate the QoE based on objective metrics requires quality assessments from the users, e.g., collected in user studies. In this context, the study design and the collection methodology might affect the view on the users' perception and the relationship between the subjective and objective metrics. Thus, a QoE model learned from pull-based QoE might differ from a push-based model. Such differences might originate from motivational factors, e.g., the continuity of providing ratings, and aspects related to the user behavior, e.g., reporting only performance issues with a self-motivated approach. This section investigates differences of a pull and push system by comparing motivational aspects, i.e., response rates and inter-arrival times of ratings, and differences in the provided ratings, i.e., ration of positive and negative assessments. Based on the results of these analyses, conclusions regarding the provided view on the QoE are drawn.

4.3.1 Analysis of Users' Motivation

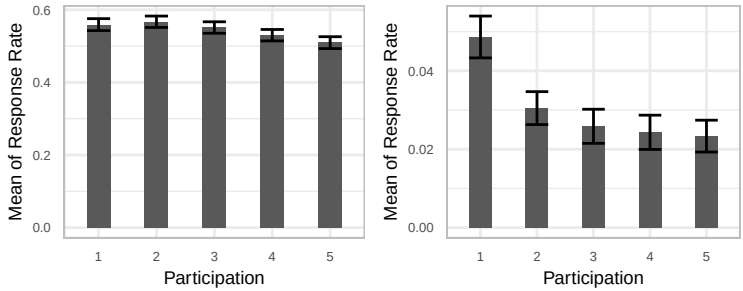
One of the main differences between the pull- and push-based collection of user ratings is that the pull approach calls the attention of the employees to the survey. In contrast, the push approach depends on the motivation of the employees which may change over time. This might result in a decrease of response rates, affecting not only the validity of results but also the representativeness. This effect may occur within but also between the study cycles. A decreasing motivation between study cycles may be caused by participating in multiple study cycles. Due to the random selection of user groups per study cycle, 77.7 % of the employees take part in more than one cycle in the long-term study. To in-

vestigate changes in the motivation, the response rates of the invited groups of employees are analyzed and compared for both pull and push-based systems.

A limitation of the study may result from using both systems in parallel leading to an unintended influence of the pull on the push system. For example, a missed, but noticed pull request might result in a push rating later on. However, only 4 % of the push ratings occur within 5 min after a missed pull request. As it is unclear if this is caused by chance, this effect is neglected in the further evaluations.

For the analysis, only users with at least two pull requests during a study cycle are considered. This excludes users who sign out from the study after the first rating request. For the remaining users, the response rate of the pull approach is defined as the ratio of given ratings and all requested ratings including the missed ones. The resulting response rate is on an hourly basis and considers only time spans in which the users were logged in the system. The response rate for ratings collected with the push approach is defined as the ratio of working hours containing ratings and those hours, in which the users were logged in the system but did not submit a rating. Again, log-in times are derived from the monitored pull requests.

Starting with a comparison of the response rates per study cycle, a trend toward decreasing rates can be observed over time. Indicated by a simple linear regression the decrease of the response rates is larger for the pull approach than for the push system. To further understand this effect, the response rate of the participants during each study cycle is analyzed. Therefore, the response rate for each day of a cycle is computed. While no evidence for a decrease of motivation when using the pull system is observable, there is a decrease in the response rate of the push system during the two weeks of a cycle. This observation is supported by a strong, negative correlation ($\rho = -0.964$) between the response rate and the day of the study cycle. Thus, the participation does not only decrease between the study cycles, but also within the cycles. To evaluate the influence of participating in multiple study cycles, the response rates



(a) Pull approach *(b) Push approach*

Figure 4.4: Mean of the response rate of employees participating in five study cycles with 95 % confidence intervals.

of consecutive study participations are evaluated. The analysis is limited to 678 employees who participated at least in five study cycles.

Figure 4.4 shows the average response rate of the first and the following four participations with 95 % confidence intervals. The left figure presents rates observed for the pull approach and results for the push-based system are shown on the right. The number of participation is depicted on the x-axes and the y-axes show the mean of the response rate. One should note that the scales of the y-axes differ for both systems. As assumed, the response rates decrease for both approaches over time. The comparison of the rates for the pull system shows that the rate is stable for the second participation and then starts to decrease, see Figure 4.4a. The significance of the differences in the response rates is established by Friedman's test for repeated measurements ($\chi^2(4) = 93.223$, $p < 0.001$). In contrast, the response rate observed for the push approach decreases immediately after the first participation and then stabilizes at a certain level, see Figure 4.4b. By using Friedman's test for repeated measurements the significance of the differences is established ($\chi^2(4) = 256.99$, $p < 0.001$). Reasons for the stabilization of the rates might be running both systems in parallel.

Requesting ratings with the pull approach keeps the employees' attention on the study. Thus, using a hybrid system might be the most efficient approach to collect self-motivated ratings on a long-term. By adapting the polling interval of the pull system, decreases in the motivation to provide pull-based ratings might be prevented. Here, adaptations of the polling interval of the pull system could be a counteract on changes in the motivation to provide push-based ratings.

Regarding the decline in the motivation to provide pull-based ratings, it is still unclear if the motivation will be continuously on the decline or if it will level out at a certain response rate. If it decreases continuously, the average rates can be predicted with a simple linear regression model learned from the observed, average response rates of the study cycles. The study duration in terms of number of conducted cycles accounts for 46.96 % of the variation in response rates with a regression coefficient of $\beta = -1.36 \times 10^{-3}$. By using the model, an average response rate of 0.528 is computed for the last cycle of an one year study. The rate declines to 0.387 after five years. This indicates that collecting pull-based ratings from employees is possible for study periods of a few years, but the analysis also highlights the importance of keeping participants motivated in long-term studies.

4.3.2 Temporal Assessment of Rating Behavior

Proceeding from the motivational aspect, the rating frequency is also an important factor when analyzing the view on the QoE derived from pull- and push-based ratings. To investigate this temporal rating characteristic, the inter-arrival times of the ratings from both systems are analyzed. Figure 4.5 shows the cumulative distribution functions (CDFs) of the inter-arrival times of ratings arriving in each system, both independent and dependent of individual users. The x-axis presents the inter-arrival time of the ratings in minutes and the y-axis shows the cumulative frequencies. Inter-arrival times of pull- and push-based ratings are presented in brown and green, respectively. The temporal characteristic is

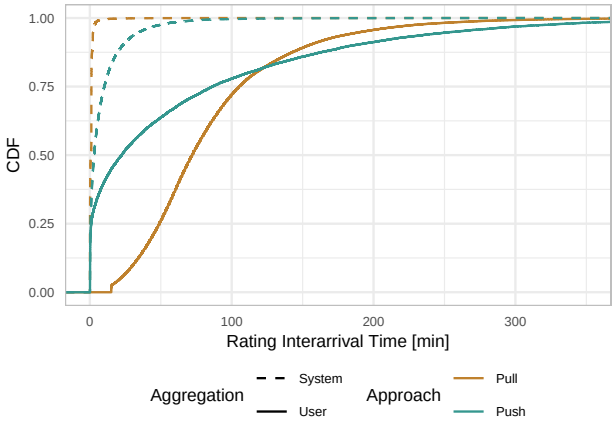


Figure 4.5: Inter-arrival times of ratings arriving in the system independent and dependent from the users.

analyzed from a global view on the system (dashed lines) as well as individually for the users (solid line).

The user-independent inter-arrival time of pull ratings is short with a mean of 0.79 min and 99 % of the ratings have an inter-arrival time of less than 6 min. In contrast, the mean inter-arrival time of user-independent push ratings is higher with 8.35 min and the 99 % percentile is about 69.58 min. This shows, that pull ratings arrive more frequent in the system than push ratings. The observation is in line with the higher response rate discussed in Section 4.3.1.

By having a closer look at the user-dependent inter-arrival times, the individual rating behavior of users who submit at least two ratings at the same day is analyzed. This includes 4 103 employees using the pull system multiple times a day, resulting in 239 983 inter-arrival times. The amount of users submitting multiple push ratings a day is lower with 1 401 employees and 10 435 inter-arrival times.

Self-motivated ratings occur more often within a shorter time interval than pull ratings, indicated by the solid, green curve which is located to the left of the solid, brown curve of pull inter-arrival times. About 13 % of the push ratings occur within 1 min after another rating. In contrast, the shape of the CDF of pull rating arrivals is nearly linear within a time span of 15 min to 120 min. This behavior is caused by the configuration of the pull system to collect ratings uniformly random once per hour.

A further analysis of limitations of the results due to running the push and pull system in parallel, brings an unexpected phenomenon to light. 4 % of the push ratings occur within a distance of one minute after an answered pull rating. Reasons may be the intention to correct a given rating or to add an additional reason for the pull rating. Hence, these ratings are excluded from the further evaluation in this chapter.

Overall, the evaluation of the rating frequencies confirms that the pull ratings provide a continuous view on the perceived quality of a user, while push ratings give a more concentrated, detailed view on short time scales. Thus, building a QoE model from push-based ratings is more sensitive towards performance changes and issues on a short-term. In contrast, the pull-based QoE model is less affected from fluctuations and allows the prediction of the QoE in a more holistic and representative manner. In both cases, building a representative model for the enterprise application requires the aggregation of the individual ratings. Due to the temporal characteristic and the sensitivity regarding fluctuations in the application performance, the push-based ratings should be aggregated on a short time intervals, e.g., hours. Otherwise, the temporal effects might be averaged out or the QoE might be dominated by these fluctuations. To predict the QoE on larger time scales, a model learned from the pull-based ratings is more suitable. These ratings are less prone of short-term fluctuations.

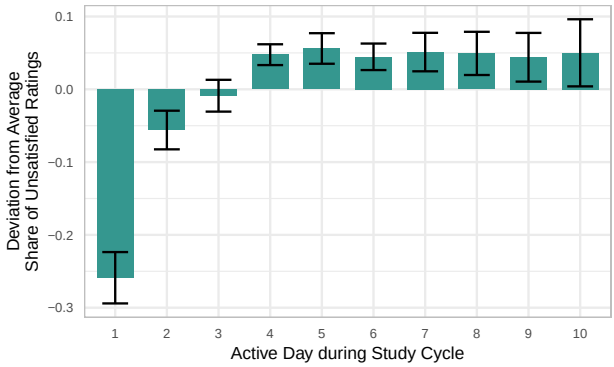


Figure 4.6: Deviation from average share of unsatisfied ratings.

4.3.3 Analysis of Rating Opinions

As the users' opinion derived from the ratings may differ between the approaches, the share of negative ratings is analyzed. While only 17.9 % of all gathered pull ratings are negative, the share of unsatisfied push ratings is higher with about 88.7 %. Other than expected, the employees use the rating tool not only to report performance issues. However, the trend toward a complaint tool cannot be denied. Reasons for that may be that users are more motivated to report performance issues than stating that everything works fine. An indicator for this hypothesis is the decreasing response rate during the study cycles observed for the push system. To establish the hypothesis, the changes in the push rating behavior are evaluated with respect to the share of unsatisfied responses during each study cycle. The evaluation is based on the activity of the participants, meaning that each day with at least one push or pull rating is classified as an active day for that user. This allows to compute the share of negative push ratings per active day for all users.

Figure 4.6 shows the mean deviation of the unsatisfied share from the mean share of unsatisfied ratings per cycle including 95 % confidence intervals. On the

first two active days, the mean deviation is negative, meaning that more satisfied ratings arrive than on average. On the third day, the share is nearly equal to the average share of negative ratings of the cycles. Day four to ten deviates positively with a higher share of negative responses than on average. The significance of the differences in the deviation is revealed by Friedman's test for repeated measurements ($\chi^2(9) = 92.827$, $p < 0.001$). A pairwise comparison using Nemenyi's multiple comparison test with Bonferroni-Holm correction reveals that the share of unsatisfied ratings is significantly lower on the first and second active day than on all other active days ($p < 0.01$). The average behavior on the other days does not differ significantly ($p > 0.05$). This results corroborate the assumption that the motivation to provide satisfied push ratings decreases over time. Running cycles with a duration of more than two weeks would converge the push approach to a pure complaint system.

Regarding the answers about affected system components given in the second step of the survey, the behavior of the participants also differs. Users gave more specific feedback about the affected system components while using the push approach. Here, the option *other* was less often selected (push: 9 %, pull: 28 %). A chi-squared test establishes that the distributions of the selected reasons differ significantly ($\chi^2(6) = 3\,266.2$, $p < 0.001$).

To sum up, with a decreasing motivation to provide positive push ratings, the self-motivated ratings converge towards a complaint system. However, the push approach provides a more specific understanding of negative ratings than the pull approach. If employees are motivated to rate the performance, their ratings occur often in short time spans, leading to a detailed but spotty view on the QoE. In contrast, the view on the users' QoE with pull ratings is more steady, but also coarser-grained per user.

4.4 QoE Modeling Based on Requested Quality Assessments (Pull Approach)

In this section, the correlations between user-provided ratings collected with the pull approach and objective metrics are analyzed. By leveraging the results of the correlation analysis a threshold-based model is designed and evaluated. This model is capable of estimating user satisfaction levels based on response time measurements while considering noise and inaccuracies caused by measurements in real-world deployments.

4.4.1 Correlation Analysis

This section focuses first on the correlation between QoE and QoS data aggregated on different levels. For the found correlations, side effects on the user's perception, i.e., seasonal effects and expectations resulting from working in different areas of the business, are evaluated.

Relationship between Pull Ratings and Performance Parameters

For estimating the QoE of the employees based on objective metrics, technical performance parameters which are most related to the user's perception of the application's quality need to be identified. Collecting ratings with the pull approach makes it difficult to map performance parameters and user-provided ratings. As the pull-based ratings were requested independently from the employees' interaction with the application, it is unclear which previous interactions lead to a specific assessment. Further, the analysis of the data on a per-user level was not allowed due to privacy policies of the company. To solve these challenges, the data is aggregated across users on two levels – per hour and per day. To do so, six descriptive statistics, i.e., mean, median, minimum, maximum, 10 %-, and 90 %-percentile, are used for aggregating each technical parameter contained in the performance data. The aggregation of the quality ratings is done by computing the share of satisfied ratings per day and per hour. Then

Pearson's correlation coefficient is computed for the user-independent share of positive pull-based ratings and all aggregated performance parameters, e.g., total response time, server processing time, network delays, number and size of packets. The evaluation is limited to days and hours each containing ratings collected from at least 25 different employees.

Despite the aggregation of the data and the varying time intervals between consecutive pull ratings, a significant negative correlation for two parameters, i.e., the mean total response time and the mean server processing time, can be observed. A negative correlation means that short average response times and server processing times lead to more satisfied ratings. The correlation coefficient for the mean total response time on a daily aggregation level is −0.46 and −0.36 on an hourly basis with $p < 0.001$. Nearly identical values are observed for the average server processing time. An explanation of this observation is a strong correlation between these two parameters (Pearson: 0.99), because the response time, composed of network delay, server processing time, idle time and other delays, is dominated by the server processing time. Although significant correlations are visible for some of the other parameters, e.g., the maximum amount of bytes sent by the server with a coefficient of −0.22, these correlations are considerably smaller. This supports the conclusion, that regarding system-related factors the QoE is mainly impacted by the waiting time. The total response time of the application is the technical parameter which approximates this waiting time the best. Observing the strongest correlation for the mean of the response time might be unexpected when considering known effects, such as the peak-end effect, while rating the quality after multiple interactions with an application. The peak-end effect describes the assessment of an experience mainly based on its peak and end, instead of judging it based on every moment [169]. However, such an effect is not visible in the correlations of parameters representing peaks in the total response time, e.g., 10 %- and 90 %-percentiles. Reasons for this absence might be the aggregation of the data, which might blur such effects. Hence, the further analysis in this section focus solely on the average total response time of the enterprise application.

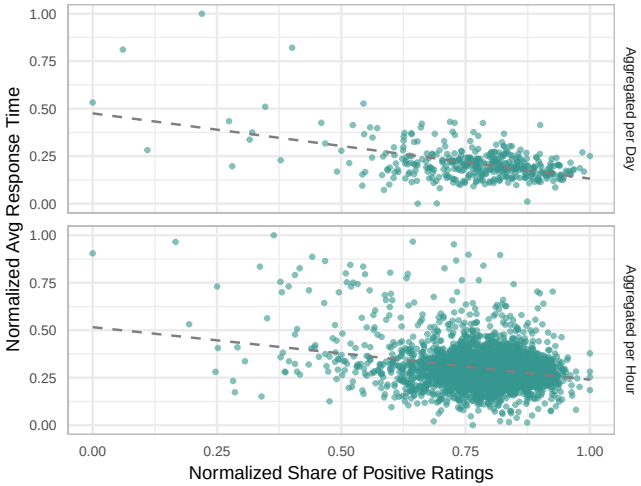


Figure 4.7: Correlation of pull ratings and response times aggregated each per hour and per day.

Having a closer look at the results for the daily and the hourly based aggregation, the correlation on an hourly basis is weaker. A reason for this effect might be that aggregating the measurements per hour results in fewer samples per aggregation interval and therefore, a larger overall variation. The variation is visualized for the mean total response time in Figure 4.7. The figure shows the aggregated average times on the y-axis and the share of positive ratings on the x-axis. The scales on the axes are normalized since the absolute values are restricted to company-internal use only.

Comparing the upper and lower scatter plot, showing the aggregation per day and per hour, respectively, the higher degree of dispersion is clearly visible. The higher variance effects also the fit of the linear model represented by the dashed line. The model explains more variation in the share of positive ratings on a daily basis ($F(1, 396) = 106.7$, $p < 0.001$, adjusted $R^2 = 0.210$) than on an hourly

aggregation level ($F(1, 3\,841) = 580.8$, $p < 0.001$, adjusted $R^2 = 0.131$). Hence, due to the lower correlation, the larger variation, and the fewer samples in the context of hourly aggregation, which might also affect the representativeness of the results, the further evaluation focuses on the data aggregated per day.

Additional Influence Factors on the QoE

The found correlation for the average daily response time and the share of positive pull ratings may suffer from ratings that are affected by system-independent, and -dependent side effects. System independent factors might be seasonal effects such as vacation times. During vacation times more employees are absent from work. This might have a positive or negative effect on the QoE. On the one hand, a smaller number of users may result in less system load and thus, the system is faster and the employees are more satisfied. On the other hand, it may lead to a higher workload and more stress for the remaining employees which might affect the perception of these users negatively. Factors related to the system or application might be changes in the regular system behavior caused by, for example, software updates, releases of new software versions, or changes in the IT infrastructure. Due to such changes the response times are no longer comparable within the whole study period of 1.5 years. Changes in the system behavior might also affect the expectations of the employees. Furthermore, working on different tasks and using different modules and components might be relevant as well. The application performance might depend on used components, e.g., functionality for customer services or sales, and thus, the expectations and perceived performance differ between groups of employees. Especially, task characteristics, e.g., complexity and completion length, in combination with the employees' expectations might affect the tolerance of waiting times [99]. To evaluate these aspects, effects caused by vacation times are analyzed. Then the influence of system changes are investigated and finally, differences in the perception of employees from three areas of the company are discussed.

Table 4.1: Correlations between the share of positive pull ratings and average total response times monitored for the same version of the application.

Period	Correlation coefficient
Mid-Nov. 2017 – mid-Dec. 2017	−0.69*
Mid-Dec. 2017 – mid-Jun. 2018	−0.64***
Mid-Jun. 2018 – mid-Dec. 2018	−0.23
Mid-Dec. 2018 – mid-Jun. 2019	−0.58***
Mid-Jun. 2019 – mid-Aug. 2019	−0.66*

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

To evaluate the hypothesis of seasonal side effects, the correlation between the user-provided pull ratings and the daily average total response times is investigated while excluding data collected during times of school vacations. This results in a slightly stronger, significant negative correlation (Pearson: −0.48, $p < 0.001$). This indicates that there are small effects caused by vacation times. To prevent even such small effects, the data collected during vacation times are removed from the data set for the further evaluation of other influence factors.

Regarding the hypothesis of influences due to changes in the system behavior, information about software releases provided by the company are considered. The software releases take place semi-annual. Thus, Pearson's correlation coefficients are determined for data collected from the same software version.

The correlation coefficients are shown in Table 4.1. Considering effects caused by releases of new software versions strengthens the correlations for all time intervals except one half-year (mid-June to mid-December 2018). During this time span, on some days the employees are satisfied even if high response times are measured and on other days fast response times lead to a large amount of negative ratings. Reasons for this observation might be changes in working processes and interfaces which are not reflected by the monitoring data. Further, an overall speed up of the application due to a software update might lead to better ratings in the first days after the update. Then, the expectations of the employees might adjust to the faster system behavior. Now, slower response times which

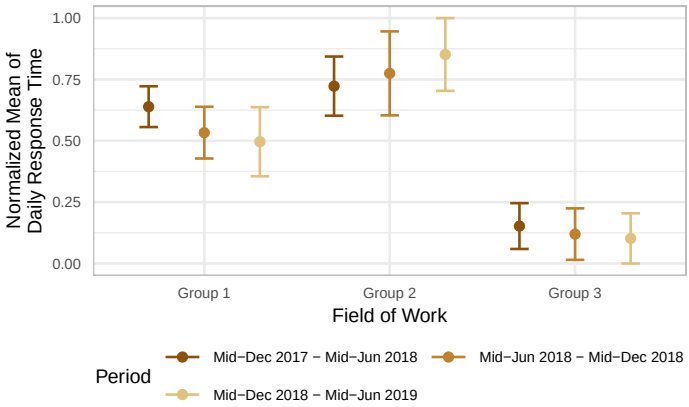


Figure 4.8: Normalized mean of daily response times including 95 % confidence intervals for three groups of employees working in different working fields.

were former perceived as fast enough might lead to a larger amount of negative ratings.

The influence of working in different fields is evaluated by analyzing the relationship of the user-provided ratings and the average total response time per day for employees from three business areas, e.g., sales and costumer services. First, the system behavior in terms of average response times measured for these employees are compared. The evaluation focuses on the three half-years between mid-December 2017 and mid-June 2019.

Figure 4.8 shows the average daily response time including 95 % confidence intervals. Again, the response times are normalized on the y-axis. The x-axis depicts the three business areas, further called working groups. The different colors define the three half-years. Comparing the working groups, the average response times differ significantly. For Group 2 the slowest average response times are observed, for Group 3 the fasted, and Group 1 lies between. By considering the confidence intervals, this observation for Group 1 is only significant

for response times measured in the last half-year of the study period. Reasons for the differences in the response time might be caused by the characteristics of the working processes. Further, the amount of processed data might be an explanation. This aspect would explain the slightly increased average response times for Group 2 when comparing the three half-years. This group works mainly in the area of customer service. The amount of data associated with customers increases over time and thus, time needed to process the historical data lead to additional delays.

Next, the influence of the observed differences in the system behavior on the satisfaction of three working groups are evaluated. Due to the privacy policies of the company which prohibit the analysis of data provided by individual users the following user-independent metric for satisfaction is defined. The system performance is assessed based on the aggregation of all ratings collected during a day. Thus, users are considered to be satisfied if a significant daily share of positive performance assessments is observed. In this work, a minimum value of 80 % positive ratings is used. This means the application performance is considered to be good if a share of at least 80 % positive pull-based ratings are provided during a day. The value was defined by the cooperating company, but can be adapted to reflect the preferences and specific characteristics of other businesses. Based on this definition, the average response time of days with a good and bad application performance is compared for the working groups. Here, only days with ratings provided by at least ten participants of the same group are considered.

Figure 4.9 shows the normalized average response time per day including 95 % confidence intervals. The response times for the three half-years are vertically stacked and values for good and bad performance are depicted in different colors. Regarding the results, three observations can be made. First, good application performance depends on the response time experienced usually by the employees. Employees of Group 1 and 2 are used to slower average response times, see Figure 4.8, hence, they are satisfied with longer waiting times compared to Group 3. Second, changes in the system behavior lead to changes in the expec-

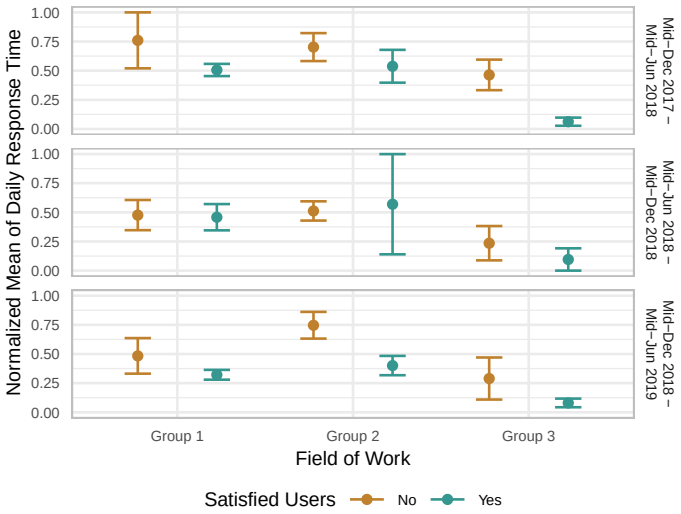


Figure 4.9: Normalized mean of daily response times including 95 % confidence intervals separated by days with good and bad application performance for three working groups.

tations and perception of the employees. This effect is seen for Group 1 when comparing the mean total response time associated with a good application performance measured in the first and third half-year. Third, for some time spans the mean response time does not differ for good and bad performance. This phenomenon is especially true for the second half-year. Here, the confidence intervals overlap for all groups. A similar effect can be observed for the perception of Group 2 during the first half-year. This behavior was expected based on the results from the correlation analysis independent from the working groups. The observation is also supported by the Pearson correlations between the average total response time and the share of positive ratings per day, presented in Table 4.2.

Table 4.2: Correlation between normalized share of positive pull ratings and normalized average response times.

Period	Field of work		
	1	2	3
Mid-Dec. 2017 – mid-Jun. 2018	-0.47^{***}	-0.25^{**}	-0.76^{***}
Mid-Jun. 2018 – mid-Dec. 2018	-0.22	-0.02	-0.31^{*}
Mid-Dec. 2018 – mid-Jun. 2019	-0.48^{***}	-0.60^{***}	-0.58^{***}

$^{*}\ p < 0.05$, $^{**}\ p < 0.01$, $^{***}\ p < 0.001$

For Group 2 the correlation coefficients of the first two study periods are small. A similar, but less strong, effect is seen for the other two groups for the second half-year of the study. This indicates, once more, that there are additional influence factors, not covered by the monitoring data, especially for Group 2.

To sum up, the correlation analysis showed a significant negative correlation between the average total response time and the user-independent share of positive ratings. These correlations were larger when taking seasonal effects and changes in the system behavior into account. Especially, the consideration of changes in the system behavior improved the correlation significantly. The analysis also revealed that working on different tasks which require different parts of the application affects the perception of the users. This effect might result from differences in the system behavior, but also expectations about the application behavior might play an important role. Furthermore, the expectation and thus, the perception of the users adapt to changes in the system behavior. Thus, considering such additional influence factors might result in a better foundation for developing an estimation model capable of leveraging correlations. However, even if such side effects are considered, a perfect correlation is not possible due to the fact that measurements in real-world deployments can be subject to noise and measurement inaccuracies.

4.4.2 Threshold-Based Model

From the perspective of the provider of the business application or the employer, the goal is to find a threshold for the mean total response time which indicates whether the employees are satisfied or unsatisfied. The definition of a good application performance (good QoE) is, again, a share of positive ratings of at least 80 % per day. This leads to a binary classification problem. Based on the average daily response time and the threshold the model classifies the performance as good or bad. Days with average response times below the threshold are labeled with a good, satisfying performance, days with measured times above the threshold are classified as bad performance days. However, measurement inaccuracies, noise, and the data aggregation can have a negative effect on the classification performance of models that rely on a single response time threshold, especially for response times near this threshold. Hence, a model that features two bounds and excludes a subset of the data in order to increase the accuracy on the remaining data is proposed. In particular, applying the two bounds to a given day leads to one of three cases. First, days on which the mean response time is above the upper response time threshold are classified as days with a low application performance and thus, low user satisfaction. In an analogous fashion, days having a mean response time lower than the lower threshold are expected to have a high user satisfaction. Finally, no reliable classification can be performed on days on which the mean response time falls between the two thresholds. By choosing combinations of thresholds that are appropriate for the given use case, it is possible to balance the accuracy increase on the data lying outside of the interval against the size of the area in which no statement can be made regarding the user satisfaction. This results in a trade-off between the fraction of data for which no reliable estimation is possible and the gains in terms of accuracy.

By using data collected during the first half-year of the study period, the trade-off is quantified. To minimize seasonal side effects times of school vacations are excluded from the data set. In addition, using only data collected from the

same software version minimizes effects resulting from changes in the system behavior.

As the data set is highly imbalanced concerning the amount of days with a share of positive ratings above or below 80 %, the balanced accuracy for the evaluation of the thresholds is applied. The balanced accuracy is computed as the average of the true positive rate (recall or sensitivity) and the true negative rate (specificity). The identification of appropriate thresholds is done on the data measured during the first two month of the study (end of November 2017 until end of January 2018). For all possible combinations of upper and lower bounds, the trade-offs between the width of the interval that is covered by the two bounds, the resulting fraction of measurement data that can not be classified in a reliable fashion, and the balanced accuracy in the remaining data set is analyzed. To this end, combinations that are Pareto optimal with respect to these three characteristics are determined. This means that there are no alternative combinations that are at least as good with respect to all characteristics and strictly better with respect to at least one.

Figure 4.10 displays these trade-offs. While the x-axis denotes the fraction of measurement data that is excluded from the classification, the y-axis represents the balanced accuracy that is achieved in the data set resulting from the corresponding combination of upper and lower bounds. Furthermore, the size of individual points is proportional to the width of the respective interval. Three main observations can be made. First, there is a high correlation between the interval width and the share of excluded data points (Pearson: 0.95). This behavior is in line with the expectation that a wider interval leads to a larger fraction of covered data. Second, when using an interval width of 0, the two-threshold model degenerates to a single-threshold model. Consequently, none of the data is excluded and the fluctuations in the measurements have the largest impact, leading to the lowest accuracy. This constellation is represented by the bottom left point. The corresponding threshold is about 826 ms with a balanced accuracy of 0.82. Finally, several gaps and plateaus can be observed in the Pareto frontier. These reflect trade-offs in which a significant amount of accuracy can

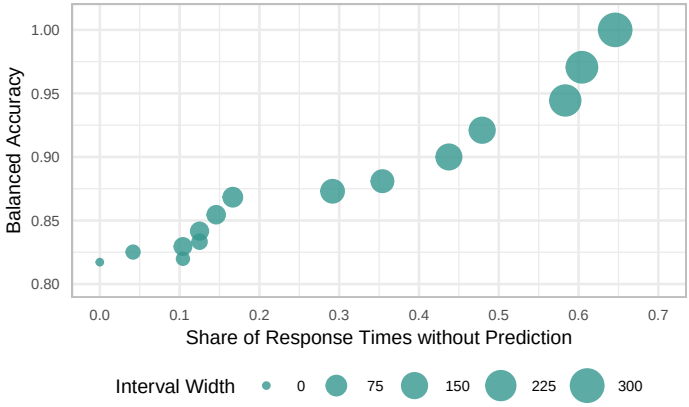


Figure 4.10: Pareto comparison of trade-offs resulting from different combinations of lower and upper bounds.

be gained for few excluded samples. For example, excluding 18 % of data results in a more than 10 % increase in terms of accuracy.

Given the accuracy requirement of 90 % which is provided by the cooperating company, the Pareto analysis from Figure 4.10 can be utilized to identify viable thresholds. These cover the interval ranging from 790 ms to 938 ms. It is worth noting that these thresholds are in the same order of magnitude as those identified in [170]. Although the context of the latter is on mobile applications that are used in daily life and a different mechanism is utilized for deriving QoE estimations, further investigations might identify a general underlying principle.

To validate the applicability of the thresholds over time, the identified thresholds with 90 % balanced accuracy are applied for data collected in the time period starting from February to mid-June 2018. Figure 4.11 shows the results of predicting days with good QoE indicated by at least 80 % positive quality ratings. The mean total response time per day is depicted on the y-axis and the share of satisfied ratings is shown on the x-axis. The vertical dashed line represents

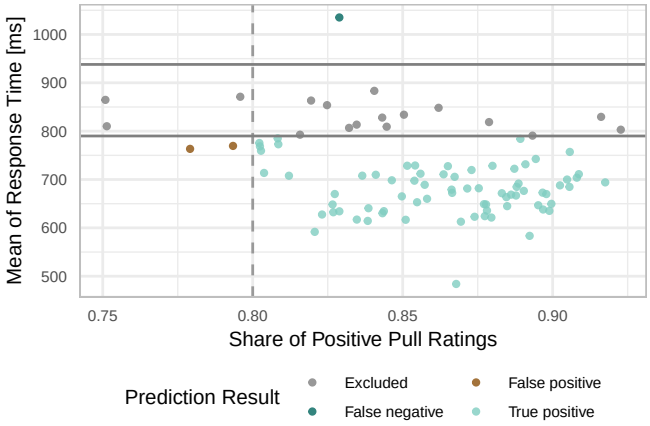


Figure 4.11: Performance of pull-based two-threshold model for data collected between February and mid-June 2018.

the 80 % threshold distinguishing between good and bad QoE. The upper and lower bounds are drawn as solid lines. Points in light colors represent correct classified days, dark colors highlight cases of misclassification.

Overall, for 18.3 % of the data no estimation is possible, as the data is covered by the threshold. This share is slightly higher than for the period between end-November 2017 and end-January 2018. Regarding the accuracy, the thresholds work very well for the class of good QoE with an accuracy of about 0.99. However, both days with a bad QoE which are not excluded from the estimation are classified incorrect, leading to a low balanced accuracy. Even if the thresholds are still applicable with an outstanding accuracy for the good QoE class, considering temporal changes in the measured response time would lead to better results in terms of balanced accuracy and amount of excluded data. Such time-dependent influence factors can be considered by using a sliding window approach for adapting the thresholds over time. As task-dependent factors also

play an important role, building individual threshold-models for the different working fields in the company might also improve the model performance. Nevertheless, the developed two-threshold-based model is a practicable strategy to predict the QoE based on noisy data collected in uncontrolled environments.

4.5 QoE Modeling Based on Self-motivated Quality Assessments (Push Approach)

This section focuses on modeling the QoE based on self-motivated ratings. The self-motivated quality ratings are collected with the push system, which runs in parallel to the pull approach. Due to the characteristics of self-motivated ratings, such a model allows the estimation of the perceived quality of performance fluctuations on shorter time scales than the pull-based model.

For identifying relevant performance parameters, the correlations between the self-motivated quality ratings and technical parameters are analyzed on different aggregation levels. Based on the results of this analysis, a machine learning-based model is developed to predict the QoE on a binary scale. Further, the performance of the model is evaluated and compared with the performance of the two-threshold-based approach.

4.5.1 Correlation Analysis between Self-motivated Ratings and Objective Metrics

First of all, the correlations between ratings collected with pull requests and push approach aggregated per day are investigated. This analysis is limited to days with at least ten ratings gathered with each approach. Considering Spearman's rank correlation ρ between the daily share of negative ratings from both approaches, a significant positive correlation $\rho = 0.433$ can be observed. Supporting the hypothesis that the employees use the push approach similar to a complaint system, the correlation can be increased to $\rho = 0.592$ by interpreting hours without any push rating as time slots where the users were satisfied.

Here, only time slots with pull assessments are considered to guarantee that the participants were able to submit ratings. The larger correlation suggests that, in contrast to the pull approach, the evaluation of the push approach should focus on the negative ratings, which might point to annoying or unacceptable performance of the enterprise application.

For the investigation of relationships between the technical data and self-motivated negative push ratings on different levels of aggregation, the performance data of different transaction types measured during the same interval of 5 min for each user is aggregated. Again, for the aggregation six descriptive statistics – mean, median, minimum, maximum, 10 %-, and 90 %-percentile – are used. Regarding the quality assessments, each 5 min interval is marked if a negative push rating was given by the user. The correlation between these Boolean indicators whether a negative push rating was given and the performance metrics is analyzed by using the point-biserial correlation coefficient. The resulting correlation coefficients for the technical parameters are all close to 0. The highest correlations can be observed for the mean of the maximum server processing time, however, the correlation is negligible ($\rho = 0.017$), whereas the correlations to the other technical parameters are even closer to zero. As the correlations are so low, which was expected, the technical data and rating data is aggregated into intervals of one hour. When considering these 5 433 aggregated intervals, the highest correlations can be observed for the mean of the minimum server processing time. It has a significant positive correlation to the share of unsatisfied users ($\rho = 0.384$), while other typical technical parameters, such as the mean of the total response time ($\rho = 0.282$), show a lower correlation. Due to the characteristics of the push-based approach, which includes short inter-arrival times (cf. Section 4.3.2), the data is not aggregated further to not lose or average out the temporal proximity of system performance and submitted push ratings. The relatively small correlation is similar to the findings for the pull ratings, an indicator that there are additional effects which are not covered by the technical data.

4.5.2 QoE Models

As the two-threshold-based modeling approach was successfully applied to estimate the QoE based on pull-based ratings, first, a similar model is developed for the push approach. Second, a machine learning-based model using multiple technical parameters to predict the QoE is introduced and evaluated.

Threshold-based QoE Model

Due to the similarity of the push approach to a complaint system, the two-threshold-based model for the self-motivated case focuses on the negative ratings. Thus, the application performance is defined by the relative number of unsatisfied users within one hour. The target threshold to distinguish an hour with a good application performance from hours with bad performance is set to 5 % of the users. This means, the performance of the enterprise application is considered to be bad if more than 5 % of the currently active users submit a negative push rating. Again, this value can be adapted to reflect the preferences and specific characteristics of other business domains and applications.

Similar to the pull-based model, the two thresholds are fitted to the mean total response time considering the trade-off between the data excluded from the prediction and the accuracy for both classes. This means, the balanced accuracy for both classes shall be maximized, and the number of hours that cannot be classified, i.e., intervals whose technical parameter lies in between the thresholds, shall be minimized. Even if a stronger correlation between the minimum server processing time and the user ratings is observed, the model is based on the mean of the total response time of the application. This allows a comparison of the thresholds and models based on pulled and pushed ratings.

When fitting the model to the data and optimizing for balanced accuracy in the first place, the two thresholds fall to the same value, which obviously also optimizes the second criterion. The resulting threshold resides at a mean total response time of 900 ms with a balanced accuracy of 0.67. The overall accuracy

is about 0.81 with corresponding per-class accuracy values at 0.84 (good QoE) and 0.51 (bad QoE).

Comparing these results to the model for pull-based QoE estimation, thresholds for bad application performance are at a similar scale despite the different aggregation levels. The less strict bound for good QoE in case of the QoE based on self-motivated ratings (900 ms vs 790 ms) leads to more false positives and thus, a lower model performance. Due to the aggregation on shorter intervals, relying a model only on the response time might not totally reflect the perception of the users regarding the waiting time. Instead considering multiple technical parameters with a machine learning-based approach might improve the prediction accuracy.

Machine Learning-based QoE Model

To develop machine learning-based models for the data of the enterprise application, a Python-based Scikit-learn[4] pipeline is used. The feature set contained in total 75 parameters. Of those, 72 parameters are objective metrics from the monitoring data. These metrics include the total response time, the server processing time, the number of sent bytes from the client and server and a Boolean indicating if an error occurred for two types of errors. The measured values are aggregated for all types of transactions and for the 20 most frequently used transactions by using six descriptive statistics. In addition to these parameters, three additional features are added as indications for the overall system load. These comprise the total number of transactions and top 20 transactions, as well as the number of actively working participants in an interval.

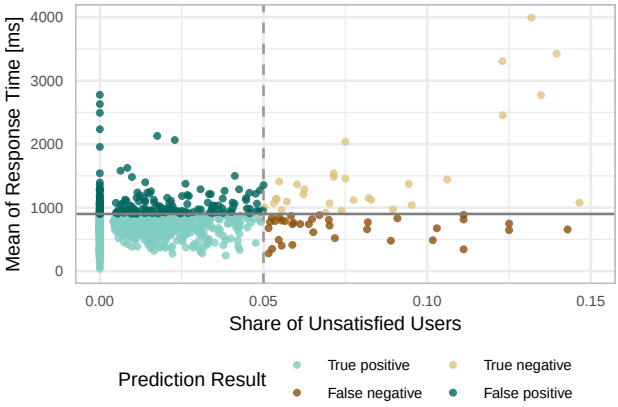
First, the 5 433 one hour intervals are randomly split into a training set of 80 % of the data, and a test set of 20 % of the data. As the class distribution (good/bad QoE) is very imbalanced, the training data were upsampled to reach an equal number of instances per class. Several feature subsets, machine learning algorithms, e.g., support vector machine, random forest, and k-nearest neighbors,

[4] https://scikit-learn.org/stable/; Accessed: August 1st, 2020

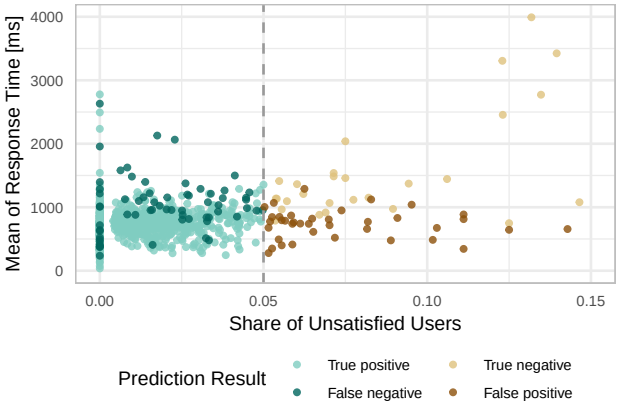
and hyperparameters were tested with a 3-fold cross-validation on the training set to select the best features, model, and model parameters. The best performing model is a Gradient Boosting Classifier with 200 regression trees using 50 of the 75 features. Its performance was tested on the test set of 1 087 intervals. Here, the prediction accuracy of the good QoE class (1 014 intervals), which is also the recall, is 0.92. The precision is 0.96, which gives an F_1-score of 0.94. For the minority class (bad QoE, 73 intervals), the precision is 0.31, the accuracy/recall is 0.53, and the F_1-score is 0.40. Thus, the performance of the machine learning-based model is better than the threshold-based model, reaching better per-class accuracy values, a better balanced accuracy of 0.73, and a better overall accuracy of 0.89.

Figure 4.12 visualizes the performance of both the threshold-based and the machine learning-based model on the test set. The x-axis shows the share of unsatisfied users including the QoE threshold at 5 % (dashed line), while the y-axis shows the mean total response time. Light green and light brown colored dots indicate correct estimations of good or bad QoE, respectively. Dark colors indicate wrong estimation, namely, false positives (dark green) and false negatives (dark brown). In Figure 4.12a, it can be seen that the threshold-based model separates the data horizontally. All intervals, which lie in the top-left sector are false negatives, erroneously classified as intervals with bad QoE. Diagonally opposite are the false positives, which are classified as intervals with good QoE although they have more than 5 % of users, which submitted negative push ratings.

In Figure 4.12b, it can be seen that the machine learning-based model overall performs better, which is expected as it is not limited to a single technical parameter. This is especially evident in the good QoE case, where almost all intervals are correctly classified. Moreover, it can be seen that the performance is also good for high mean total response times. Only in case this technical parameter is low, the model loses a lot of its discriminative power, and misclassifies several intervals into false positives. An important observation is that many of these intervals have very low values of the technical parameters, i.e., an objectively fast performance of the enterprise system. This suggests that, in these



(a) Threshold-based model.



(b) ML-based Model.

Figure 4.12: Performance of push-based QoE models on test set.

cases, possibly the technical system was not (only) responsible for the negative push ratings, but maybe other work-related issues triggered the ratings.

To sum up, both the threshold- and the machine learning-based model allow to map the self-motivated push ratings onto QoE. As the threshold-based model only considers a single technical parameter, its decision boundary introduces a lot of false positive and false negative classifications. In contrast, the machine learning-based QoE model uses all technical parameters, and consequently, can significantly reduce the false negatives. However, due to the class imbalance and possibly other non-technical issues, which are not captured by the measured data, the performance on the bad QoE class is lower than on the good QoE class. Thus, it was shown that it is also possible for the push-based approach to estimate the QoE both with simple threshold-based and more complex machine learning-based models. Again, including additional features, related to characteristics of the field of work, might improve the model performance.

4.6 Lessons Learned

This chapter focused on the estimation of the QoE of enterprise applications based on real-world monitoring data which might be subject to measurement inaccuracies. The QoE derived from pull and push-based ratings served as a foundation for the QoE model. In this context, the approaches for collecting user ratings were analyzed and compared regarding the provided view on the QoE. For both approaches, the relationship between the QoE and multiple technical parameters aggregated on different levels was investigated. Especially for the ratings collected with the traditional pull approach, additional influence factors on the QoE were taken into account. Finally, based on the results of the correlation analyses a two-threshold-based model estimating the pull-based QoE as well as a machine learning-based model for the self-motivated ratings was developed and evaluated. The data for the analyses and evaluations originate from a large user study conducted during a period of 1.5 years in a cooperating company.

The research questions, derived from gaps in the literature as outlined in Section 4.1, were answered as follows. Section 4.3 focused on the question about differences in the QoE derived from ratings collected with the traditional pull approach and the self-motivated push system. The analysis of the rating characteristics showed that the pull-based approach led to a continuous, but coarse-grained view on the QoE of the employees. Despite observable decreases in the motivation, the results remained stable and representative for 1.5 years. This demonstrates the capability to monitor the QoE on a long-term with the pull-based approach. Building a QoE model from pull-based assessments allows the prediction of the QoE on larger time scales, e.g., days, due to the lower rating frequency. Such a model can be used to estimate the perceived quality of the application performance in a holistic and representative manner. In contrast, the self-motivated ratings provided a more detailed, but spotty view on the QoE. Further, the usage of push-based system converged toward a pure complaint system, due to a loss of motivation to provide positive feedback on the application performance. Hence, a QoE model built from self-motivated ratings should focus on negative ratings. Due to the spotty characteristic of the ratings, such a QoE model is more sensitive to short-term fluctuations in the application performance. The model can be used to predict the QoE on shorter time scales, e.g., hours, than a pull-based model. However, collecting self-motivated ratings for learning or retraining a model should be done either in user studies with a reasonable duration or with a hybrid pull- and push-based system to prevent decreases in the motivation to provide complaints.

Section 4.4 addressed the questions and challenges related to estimating the pull-based QoE. Here, questions such as the presence of relations between pull-based ratings and technical parameters were answered with respect to additional influence factors on the employees' perception. The analysis gives evidence about the presence of a significant negative correlation between the daily share of positive ratings and the average daily total response time of the application. It highlighted the influence of changes in the system behavior and working in different parts of the business on the expectations of the employees and thus,

on the QoE. However, due to variances in the monitoring data a perfect correlation between the ratings and the performance parameters was not found. To handle these variances, a two-threshold-based QoE model was developed. By excluding data covered by the two thresholds, noise, inaccurate measurements, and artifacts caused by data aggregation were taken into account. This approach led to a trade-off between the amount of excluded data and an improvement of the prediction accuracy. The performance evaluation of the model also showed the importance of considering temporal changes in the technical data as well as changes in the tolerance of the users. This aspect is even more relevant when estimating the QoE of business applications as the users work with the application every day over years. Even if the identified thresholds are individual for the specific business application, the proposed two-threshold-based model is a generic approach, applicable for web- or network-based applications or services in enterprise environments.

Section 4.5 covered the research questions about relations between self-motivated ratings and objective metrics and about the possibility to estimate the QoE deduced from these ratings. The weaker correlations between the user satisfaction and the performance parameters revealed that the ratings were not only triggered by slow response times of the application. Instead, other performance parameters as well as non-technical factors might be relevant and should be considered when modeling the QoE. By focusing on the unsatisfied ratings, a machine learning-based model is trained on multiple performance related features. This model outperformed the threshold-based approach and proofed the possibility to predict the QoE based on self-motivated ratings. However, the performance evaluation also exposed that the model accuracy suffered from cases with good performance values, e.g., low response times, in combination with a high share of unsatisfied users. This again indicated that there were additional influence factors on the QoE, which were not included in the technical data. Considering non-technical influence factors seems to be even more relevant when using the self-motivated approach. The user's tolerance threshold towards performance issues might depend on contextual factors and individual

user characteristics. Including these aspects in the model, e.g., in terms of the field of work, or building individual models might be a promising approach to improve the model accuracy in future work.

5 Conclusion

Today, daily life is shaped by the use of technological achievements. This ongoing trend covers leisure time activities but also the everyday work and led to a rethinking of service and application providers. Providers broadened their focus from pure Quality of Service (QoS) oriented management processes to user-centric concepts such as the concept of Quality of Experience (QoE). Following this strategy, QoE models are used to predict the performance quality as perceived by the users based on objective performance metrics of the application. Considering the satisfaction of the users with the application's quality is even more relevant in the domain of business systems. The satisfaction of the employees with the system quality is becoming a main business driver as performance issues might reduce the efficiency and performance of the workforce.

While the concept of QoE is well studied for multimedia applications such as video streaming or VoIP services, transferring it to the domain of business applications is not trivial and leads to numerous open challenges. These challenges arise from the involvement of employees in user studies during their regular work. In addition, real-world measurements under uncontrolled conditions, e.g., in production systems, might lead to inaccuracies and unexpected side effects on the QoE. Thus, QoE monitoring and QoE estimation have to be able to handle noisy monitoring data which might only reflect partly the user's perception.

This monograph contributed to answer open research questions on the monitoring and estimation of QoE of enterprise applications. To do so, approaches and methods of conducting subjective experiments with crowdsourcing were adapted to the enterprise environment. These adaptations were described as a general concept for monitoring and modeling the QoE of business applications.

A concrete realization was shown by designing a survey tool which considers enterprise specific requirements. Furthermore, multiple user studies were conducted on microtasking platforms as well as in the enterprise environment. The studies addressed questions on the improvement of the results of crowdsourcing tasks, the impact of dimensions of the study design on the QoE, and challenges related to the QoE modeling of enterprise applications based on real-world measurements.

The research questions posed in the introductory section were answered as follows. The question on how to monitor the QoE of enterprise applications was addressed by presenting a general concept. The concept included the whole monitoring process, beginning with the identification of applications affected by performance issues, e.g., by using ticketing systems. By transferring crowdsourcing-based methods into enterprise environments to conduct subjective experiments to analyze the QoE, approaches for the evaluation of influence factors were conceptualized. Especially, advantages and disadvantages of studies conducted in dedicated test environments or in production systems were highlighted. The latter approach benefits from realistic test situations, but suffers from uncontrolled conditions. This aspect also needs to be taken into account during the analysis of the relationship between performance parameters and user-provided ratings and the creation of a QoE model.

Regarding the usage of crowdsourcing, the monograph focused on the question how to optimize the quality of results by means of the task design. To answer this question, approaches for improving the varying quality of results of tasks and studies were investigated. These approaches focused on the optimization of the task design under consideration of demands of requesters and workers. From the requester's side, this included the evaluation of the applicability of a newly developed, task independent quality control mechanism based on the attention of the workers. The evaluation showed the prediction and filter validity of this mechanism, but also revealed differences between users from different platforms. However, by adapting thresholds for the required degree of attention, the filter can be adjusted to the specific needs of different types of tasks or re-

search studies as well as to platform characteristics. Differences were also found regarding worker's preferred task properties and design elements between but also within platforms. While the workers agreed on the importance of task properties such as an adequate payment, clear task instructions, and a good usability of the task interface, no clear consensus was observed on favored interface design elements. This makes it nearly impossible to design a task which meets the preferences of all workers. Nevertheless, using optimization strategies which focus on general task properties might result in a quality improvement and still consider the workers' demands. Thus, the decomposition of complex tasks to simpler sub-tasks and a workflow allowing the task processing in native language were evaluated towards their impact on the quality of work results. The decomposition of complex tasks to simpler sub-tasks only partly improved the quality. For some kind of sub-tasks the quality was even worse due to a loss of context. Thus, decomposing a task to its maximum is not advisable. The evaluation of the second approach to simplifying tasks by lowering language barriers showed a quality improvement for tasks which require the creation of text. Providing only the instruction in native language neither showed a positive nor a negative effect. However, the results of both studies gave evidence that good task instructions reduce negative side effects on the workers' performance.

As the investigation of the QoE of enterprise applications with users without domain knowledge might lead to unrepresentative results, involving employees might be the more promising strategy. Here, similar to other crowdsourcing-based unsupervised studies the study design plays an important role as it might influence the study results. Thus, relevant design dimensions were identified and their impact on the QoE of enterprise applications were discussed. The dimensions are related to the context, the participants' domain knowledge, the artificiality of the test system, the tested parameter space, and the controllability of conditions. The impact of the artificiality of the test setup and the domain knowledge of the participants on the QoE were explored in user studies. The analysis of the impact of the artificiality of the test setup on the QoE showed that the test interface had no influence on the user ratings. However, partici-

pants tended to be less attentive while interacting with a test interface which modeled the real interface of the tested application. In addition, an unnatural behavior of the participants was observed due to the test situation. The study on the influence of the domain knowledge of participants on their perception revealed differences between experts and laypersons, especially laypersons with microtasking background.

The findings of both studies suggested that the QoE of enterprise applications should be monitored from employees during their regular work. This leads to the question how to realize such an approach. To answer the question a non-intrusive survey tool was introduced and its applicability was evaluated in user studies conducted in a cooperating company. The developed tool allows the integration of the collection of quality assessments into the day-to-day work of employees. It realizes a pull-based collection approach as well as a self-motivated system. The pull approach models the traditional method which actively requests ratings from study participants. The frequency of the requests can be adapted to the individual situation in each company. The feedback given by employees participating in a user study validated that requesting ratings once per hour was an applicable frequency in the cooperating company. In contrast, the push approach is self-motivated comparable to an error report or complaint system. It allows the employees to rate the application's quality anytime.

Using different methods to collect quality ratings posed the question of differences in the provided view on the QoE. A self-motivated rating might be triggered by the perceived quality, e.g., after interacting with the application, while the pull-based rating is triggered by the survey tool. The analysis of ratings collected with both approaches established this assumption. The quality assessments were gathered from employees in a cooperating company in a user study during a period of 1.5 years. While the pull-based approach provided a continuous, coarse-grained view on the QoE, gathering self-motivated ratings led to a spotty, but more detailed view on the QoE. However, both systems suffered from a decreased motivation to provide ratings. One reason for this observation was a decline in the motivation to submit push-based ratings in cases the

employees were satisfied with the application's performance. Hence, the usage of this rating system converged towards a pure complaint system. This aspect needs to be taken into account when creating a QoE model based on this type of ratings, e.g., by focusing on ratings reporting a bad application performance. Nevertheless, the decrease in the response time over time for both systems requires further investigation. Based on the results no final statement can be made in respect with the sufficiency of the number of provided ratings after using the tool for multiple years.

The enrichment of the subjective data with technical monitoring data of the business application mainly used in the company was basis of the analysis of effects influencing the relationship between user-provided ratings and technical performance data. A correlation analysis of the data aggregated on different levels, i.e., per hour and per day, revealed again differences between the rating methods. The results suggested to look at pull- and push-based data on different levels of aggregation. Variances in the pull-based ratings were better explained by the daily mean of the total response time of the application than by hourly values. The more detailed view on the QoE would be averaged out by aggregating the push-based data per day. Further for the push-based ratings, correlations were found between multiple performance parameters and the share of unsatisfied employees on a hourly basis. However, these correlations were weaker than observed between the share of positive pull ratings and average total response times. Independent of the rating method, the correlation suffered from measurement inaccuracies and additional effects which were not visible in the technical monitoring data. By concentrating on the pull-based ratings, the impact of changes in the system behavior and working on different business processes on the quality ratings were investigated. For employees of distinct business areas, e.g., sales and customer care, the expectations regarding the regular system behavior differed. This led to differences in the perceived application quality. Regarding temporal effects, their expectations changed over time caused by, for example, changes in the system behavior after software updates.

By leveraging the found rating characteristics and correlations, the question on how to estimate the QoE of enterprise applications from noisy real-world data was finally tackled. A two-threshold-based model overcame drawbacks from noisy data and uncovered influence factors by excluding such data from the prediction. The QoE derived from pull-based ratings was estimated based on the average daily total response time. The evaluation of the trade-off between the amount of excluded unpredictable data and the model accuracy revealed that removing a small share of noisy data led to a significant increase in the accuracy. The performance evaluation also highlighted the necessity of adjusting the thresholds to temporal changes in the system behavior. The prediction of the QoE deduced from self-motivated ratings was done with a machine learning-based approach. Here, only performance parameters were considered. Thus, model performance suffered from cases where the users' perception was influenced by additional factors. Including such factors into the model, e.g., the working field of the employees, might improve the performance significantly. However, considering all factors influencing the QoE might be not applicable for enterprise applications, e.g., due to privacy reasons.

The concept, methods, and results presented in this monograph can be used as a guideline for monitoring and estimating the QoE of enterprise applications. The developed tool as well as the proposed methods for collecting quality ratings from employees point out how to realize QoE studies in enterprise environments in a scientifically valid but still economic way. Regarding the modeling, pitfalls were found which arise in particular from the data acquisition in a real-world scenario. These cover aspects such as the decline in motivation to submit assessments and temporal effects, e.g., changes in system behavior and adaptations of user expectations to them. However, with the two-threshold-based model a simple modeling approach were proposed which is able to deal with measurement uncertainties in real-world data. Besides these practical aspects, the insights on the impact of waiting times and contextual factors on the QoE of enterprise applications gained from the long-term user study serve as a good foundation for future research on this topic.

Appendices

A Summary of Applied Tests

The following table gives an overview about the applied tests in this work. The overview includes a brief description and assumptions of each test.

Table A.1: Summary of applied tests.

Test	Usage and assumptions	Section
Pearson chi-square test (with Yates' continuity correction)	Tests the independence of two independent, categorical variables. Each observation must have an equal probability to occur with cell counts greater than 5 (for 80% of the cells for large tables) [171].	2.2.3, 2.3.3, 2.4.1, 2.4.2, 2.5.2, 3.3.2, 4.3.3
Cramer's V	Shows the magnitude of an effect shown by the chi-square test. It is applicable on contingency tables larger than 2x2 [172].	2.4.2, 2.5.2
Kolmogorov-Smirnov test (non-parametric)	Tests if the distributions of two samples are the same. Variables must be independent and at least ordinal. The test is only precise for continuous variables [173].	2.5.2
Kruskal-Wallis test (non-parametric rank-sum test)	Tests if more than two samples originate from the same population. Variables must be independent and ordinal-scaled [171].	2.2.3, 3.3.2, 3.3.3

Levene test	Tests the homogeneity of variance of independent, continuous variables [174].	2.2.3
Welch t-test (independent)	Tests if two group means are different. Assumes homogeneity of variance and normal-distributed samples. Observations have to be independent and at least at interval-scaled [175].	2.5.2
Mann-Whitney U test (non-parametric)	Tests for differences between two independent samples. The dependent variable should either be ordinal or continuous [176].	3.3.2
Wilcoxon rank-sum test (non-parametric)	Is an alternative to the Mann-Whitney U test [177].	3.3.2, 3.3.3
Wilcoxon signed rank test (non-parametric)	Tests for differences between two related samples. Assumes at least ordinal-scaled variables; Paired observations should be randomly and independently drawn [178].	3.3.3, 3.4.3
One-way ANOVA	Tests if three or more means are the same. Assumes homogeneity of variance and normal-distributed samples, independent observations, and at least an interval-scaled dependent variable [179].	2.5.2
Welch ANOVA	Is a robust version of the one-way ANOVA. Assumes normal-distributed samples, but not necessarily equal variance [180].	2.2.3

Tamhane post hoc test	Is a post hoc test for Welch ANOVA which handles unequal variance [181].	2.2.3
Friedman's test	Tests for differences between related groups (robust version of ANOVA). Assumes more than two conditions, measured with the same test group. Variables should be on an ordinal or continuous scale [182].	2.3.3, 3.4.3, 4.3.1, 4.3.3
Nemenyi's post hoc test	Compares multiple ranks of joint samples, e.g., used after significance of Friedman's test [183].	2.3.3, 4.3.3
Pearson's correlation coefficient	Characterizes the strength of the relationship between two continuous or interval-scaled variables. For determining significance of correlation the variables have to be normal-distributed [184].	2.2.4, 2.2.5, 3.3.3, 4.3.1, 4.4.1
Point-biserial correlation coefficient	Is a special case of Pearson's correlation where one variable is dichotomous [185].	2.2.5, 3.3.3
Spearman rank correlation	Is a non-parametric rank correlation used when parametric assumptions are violated [186].	4.3.1, 4.5.1
Cronbach's alpha	Tests the internal consistency of a test or measurement instrument [187].	2.2.3

B Screenshot and Questionnaires of User Studies

B.1 Using Attention Testing for Quality Assurance

Table B.1: Questions asked before attentiveWeb.

Questions	Possible answers
Please select your gender[1]	–
	Male
	Female
Please select your age[1]	–
	< 20
	20–30
	31–40
	41–50
	>50
Please select your country[1]	–
	List of countries taken from
	https://maxmind.com/
Is English [WiSoPanel: German]	–
your mother tongue?[1]	Yes
	No

[1]Questions skipped in the studies on predictive validity.

Please rate your English [WiSoPanel: German] skills[1]	–
	Beginner
	Advanced
	Native speaker
Where do you use English [WiSoPanel: German]?[1]	–
	At school/at work
	Daily life
	Vacations
Where are you at the moment?[1]	–
	At home
	At work
	Internet café
	Somewhere else
How many people are around you?	–
	0
	1–3
	4–10
	> 10
How mentally focused are you at the moment?	–
	1: Not at all
	2
	3
	4
	5: Very highly
How skilled are you with the mouse?[1]	–
	1: Unskilled
	2
	3
	4
	5: Highly skilled

Table B.2: Questions asked after attentiveWeb.

Questions	Possible answers
Please select your continent[1]	–
	List of continents
Where are you at the moment?[1]	–
	At home
	At work
	Internet café
	Somewhere else
Have you ever done this test before?[1]	–
	Yes
	No
If you did the test before, when was it?[1]	–
	< 1 week
	1 week - 1 month
	1 month - 1 year
	> 1 year
Which strategy did you use for the test?	–
	As fast as possible, accepting mistakes
	Slower but correct answers
How mentally focused are you at the moment?	–
	1: Not at all
	2
	3
	4
	5: Very highly
Did you do the test seriously?	–
	Yes
	No
Feedback	Free text

B.2 Task Selection and Design Preferences – Selected Questions

14. What kind of tasks do you usually work on? (Choose all that apply)

- ☐ Content generation (e.g. writing an article or a review)
- ☐ Data collection (e.g. finding information by searching the web)
- ☐ Content rewriting / content summarization
- ☐ Object classification (e.g. image tagging, image filtering/moderation, product categorization)
- ☐ Audio transcription
- ☐ Website feedback and advise (e.g. usability testing)
- ☐ Surveys, questionnaires, user studies
- ☐ Creative work (e.g.role plays)
- ☐ Translation
- ☐ Data Entry (e.g. image transcription)
- ☐ Proof Reading
- ☐ Other

15. What are the TOP 3 kinds of task you like the most?
(Please do not choose more than three options)

- ☐ Content generation (e.g. writing an article or a review)
- ☐ Data collection (e.g. finding information by searching the web)
- ☐ Content rewriting / content summarization
- ☐ Object classification (e.g. image tagging, image filtering/moderation, product categorization)
- ☐ Audio transcription
- ☐ Website feedback and advise (e.g. usability testing)
- ☐ Surveys, questionnaires, user studies
- ☐ Creative work (e.g.role plays)
- ☐ Translation
- ☐ Data Entry (e.g. image transcription)
- ☐ Proof Reading
- ☐ Other

Figure B.1: Questions about preferred kind of tasks usually selected on the crowdsourcing platform.

17. To which extent do you consider the following aspects to be important or unimportant when selecting a task?
Please decide spontaneously. Do not think too long about your decision to make sure that you convey your original intuition.

	Totally unimportant	Unimportant	Neutral	Important	Very important
Appropriate size of salary	○	○	○	○	○
Appropriate estimation of the time required to complete the task	○	○	○	○	○
Clear instructions	○	○	○	○	○
Easy-to-use page design	○	○	○	○	○
Appealing graphical design	○	○	○	○	○
Multiple tasks available	○	○	○	○	○
Bonus for good performance	○	○	○	○	○
High reputation of the employer	○	○	○	○	○
Ability to contact the employer	○	○	○	○	○
Task sounds fun to complete	○	○	○	○	○
Simplicity of the task	○	○	○	○	○
Challenge of the task	○	○	○	○	○

Figure B.2: Question about important/unimportant task properties while selecting tasks.

18. In common task design, to which extent do you consider the following aspects as frustrating (or not)?
Please decide spontaneously. Do not think too long about your decision to make sure that you convey your original intuition.

	Not frustrating at all	Not frustrating	Neutral	Frustrating	Very frustrating
Inadequate payments	○	○	○	○	○
Too short time allocations	○	○	○	○	○
Ambiguous or incomprehensible instructions	○	○	○	○	○
Non-intuitive/hard-to-use page design	○	○	○	○	○
Old-fashioned or poor graphical design	○	○	○	○	○
Batches that contain only a few tasks	○	○	○	○	○
No Bonus regardless of how good the performance has been	○	○	○	○	○
Employers that are not highly esteemed by other workers	○	○	○	○	○
No possibility to get in touch with the employer	○	○	○	○	○
Boring tasks without any fun factor	○	○	○	○	○
High complexity of the task	○	○	○	○	○
Task not challenging / too easy	○	○	○	○	○

Figure B.3: Question about task properties considered as frustrating.

19. For each of the following aspects, please pick the alternative that best fits your preferences.
Please decide spontaneously. Do not think too long about your decision to make sure that you convey your original intuition.

I prefer ...

- ◯ tasks that are embedded in microWorkers.
- ◯ working on tasks on external websites and copying a payment code.

Concerning the task structure I prefer ...

- ◯ a division of the task into serveral pages (e.g. one page per subtask).
- ◯ displaying the whole task on one page.

I usually prefer ...

- ◯ working on longer tasks as I do not like adapting myself to permanent changing, new tasks.
- ◯ working on short tasks as thus I can compete many tasks per hour and as it gives me some variety.

The instructions should be ...

- ◯ as short as possible to reduce the required time for reading.
- ◯ as accurately as possible to avoid misunderstandings.

Background information about the task ...

- ◯ should be provided as it helps to understand the overall intention of the task.
- ◯ do not interest me, but mean only an additional amount of text to read through.

Illustrated examples ...

- ◯ help me to understand the task and reduce task ambiguity.
- ◯ distract from the actual task and are superfluous.

Highlighting of important information ...

- ◯ misleads me to only read the highlighted part of the instructions.
- ◯ helps, but does not prevent me from reading the instructions carefully.

Within task instructions, illustrated examples are ...

- ◯ more important than highlighting of important information.
- ◯ less important than highlighting of important information.

Figure B.4: Questions about task design preferences.

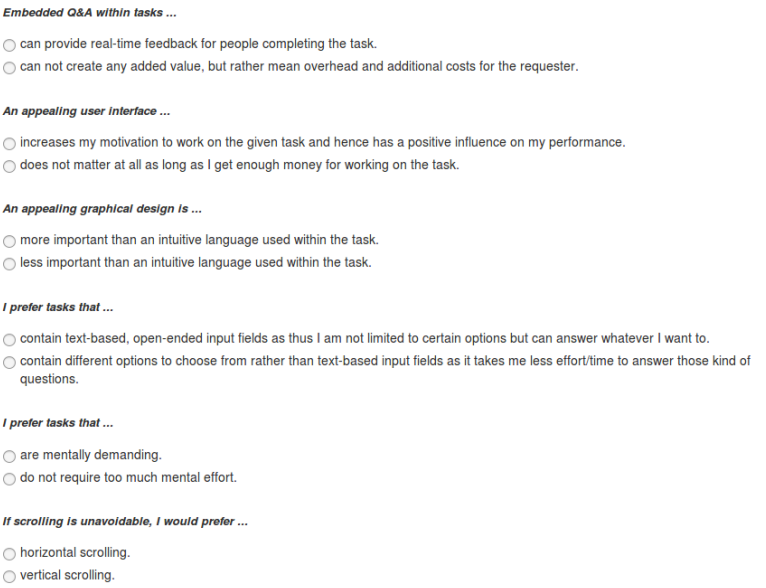


Figure B.5: Questions about task design preferences.

B.3 Impact of Task Decomposition – Screenshots of User Interfaces

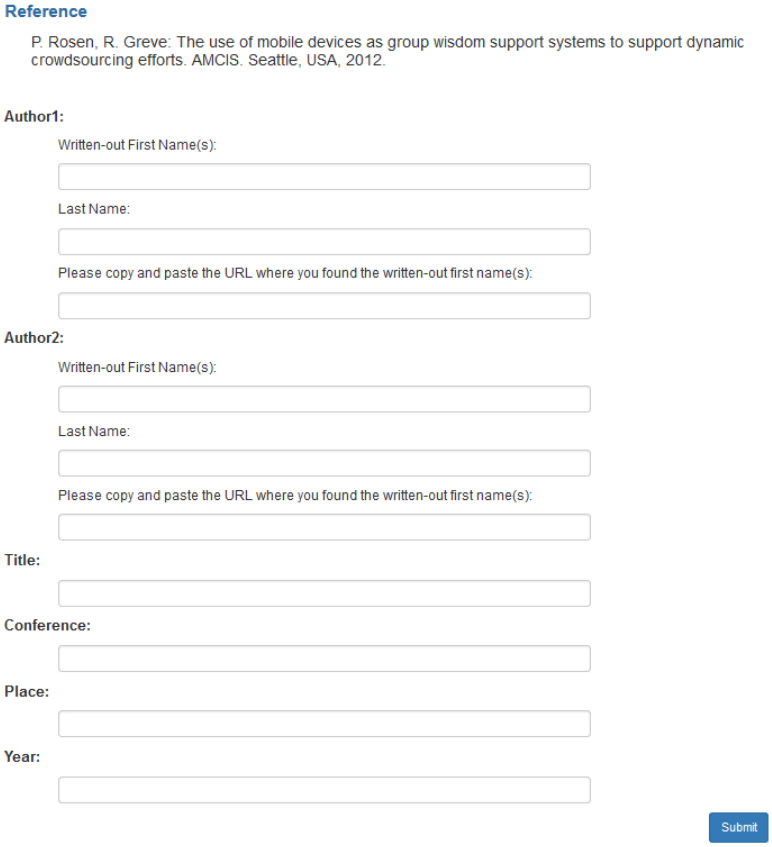


Figure B.6: Screenshot of the user interface of Task 1.

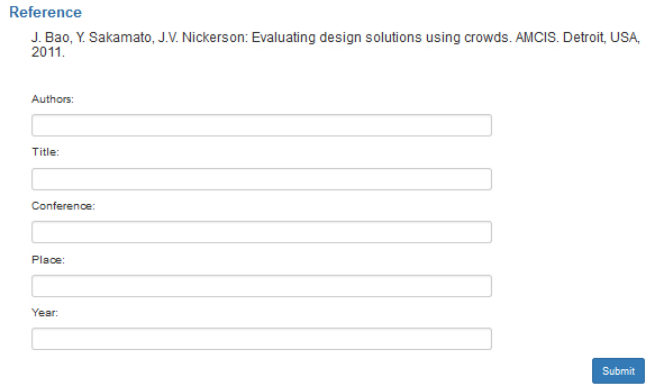


Figure B.7: Screenshot of the user interface of Sub-task 2a.

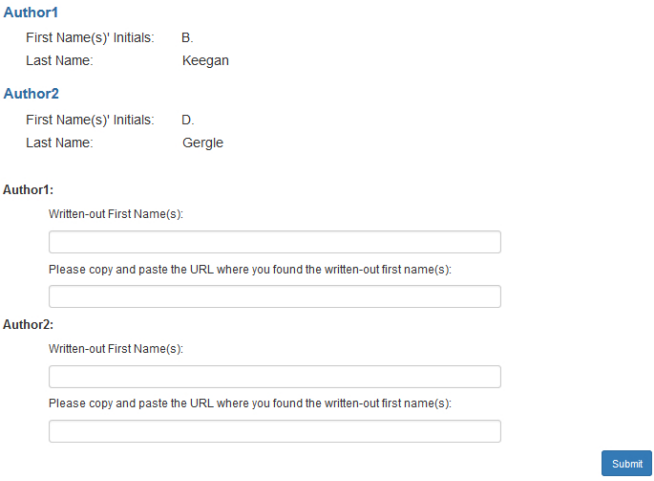


Figure B.8: Screenshot of the user interface of Sub-task 2d.

B.4 Influence of Test System Artificiality

Table B.3: Questions asked before watching the video.

Questions	Possible answers
What is your age?	List of predefined numbers (18 – 90 years)
What is your gender?	Female
	Male
	Other
	Prefer not to answer
Which continent do you live on?	Africa
	Asia
	Australia
	Europe
	Northern America
	Southern America
What is currently the highest degree you have obtained?	Primary School
	Secondary school
	Bachelor
	Master
	Ph.D
	Something else
What is your current professional activity?	Factory/service worker
	Employee/civil servant
	Self employed/free profession
	Pensioner
	Student
	Unemployed
	Housewife/househusband
	Something else

How often have you surfed the Internet during the last month?	Several times a day Once a day Several times a week Once a week Several times a month Less often Never
How often have you watched video clips/streams on the Internet during the last month?	Several times a day Once a day Several times a week Once a week Several times a month Less often Never

Table B.4: Questions asked after watching the video.

Questions	Possible answers
Did you like the video you just saw?	Extremely Fairly Moderately Slightly Not at all
What was shown in the video?	
Soccer game:	A tennis game A soccer game A baseball game
Animals:	Fishes Lions Horses
Concert:	A climbing scene

	A car race
	A pop concert
Did you notice any stops while the video was playing?	No
	Yes
How many stops did you notice?	List of numbers (0 – 10)
When did you notice stops during the video playback? (Multiple choice)	Never
	In the first half
	In the middle
	In the second half
Did you experience this stops as annoying? (If you did not notice any, stops, please select "Not at all")	Extremely
	Fairly
	Moderately
	Slightly
	Not at all
Please rate the overall quality of the video streaming.	Excellent
	Good
	Fair
	Poor
	Bad
Would you watch video clips, which have the same quality like the video that you have just watched?	No
	Yes
Additional question, if a recommended video was clicked:	
What was the main reason for clicking a recommended video?	I did not like the played out video
	The recommended video sounds interesting
	The played out video stopped
	I wanted to explore the website
	I clicked the link intuitively
	Other reasons

Table B.5: Questions asked in the final questionnaire.

Questions	Possible answers
Please select your country	List of countries taken from https://maxmind.com/
How often have you watched video clips/streams on the Internet during the last month?	Never Less often Once a month Several times a month Once a week Several times a week Once a day Several times a day
Describe your active Internet time:	Never Less often Once a month Several times a month Once a week Several times a week Once a day Several times a day

B.5 Influence of Domain Knowledge

Table B.6: Questions about personal information asked at the end of the test.

Questions	Possible answers
Please select your age (optional)	-
	List of predefined numbers
Please select your gender (optional)	-
	Female
	Male
What is your highest degree?	I did not complete high school
	Special needs school
	High school
	College
	Bachelor's Degree
	Master's Degree
	Advanced Graduate work or Ph.D
Do you work in health care?	Yes
	No
Do you have children or do children	-
live in your household in the	Yes
age between 0-6 years?	No

C Parameters contained in Monitoring Data

Table C.1: Overview about parameters included in the monitoring data.

System	Transaction
Time stamp	Type
User ID	Number of transactions
Client IP address	Total response time (ms)
Terminal server	Server processing time (ms)
Data center	Network time (ms)
Application module	Idle time (ms)
Server name	Other delays
Server IP address	Length (pkts)
	Hits per transaction
	Client transaction size (B)
	Server transaction size (B)
Network	**Errors**
Bytes sent by server	Sum of failures
Bytes sent by client	RPC errors
Packets sent by server	SAP GUI errors
Packets sent by client	SAP RFC errors
RTT server (ms)	SMB errors
RTT client (ms)	TCP errors

Bibliography and References

Bibliography of the Author

Journal Papers

[1] A. Göritz, K. Borchert, and M. Hirth, "Using Attention Testing to Select Crowdsourced Workers and Research Participants," *Social Science Computer Review*, 2019. DOI: 10.1177/0894439319848726.

[2] F. Kunz, M. Hirth, T. Schweitzer, C. Linz, B. Goetz, A. Stellzig-Eisenhauer, K. Borchert, and H. Böhm, "Subjective Perception of Craniofacial Growth Asymmetries in Patients with Deformational Plagiocephaly," *Clinical Oral Investigations*, 2020. DOI: 10.1007/s00784-020-03417-y.

Conference Papers

[3] M. Hirth, K. Borchert, K. de Moor, V. Borst, and T. Hoßfeld, "Personal Task Design Preferences of Crowdworkers," in *Proceedings of the 12th International Conference on Quality of Multimedia Experience (QoMEX)*, IEEE, Athlone, Ireland, May 2020, pp. 1–6.

[4] K. Borchert, A. Schwind, M. Hirth, and T. Hoßfeld, "In Vivo or in Vitro? Influence of the Study Design on Crowdsourced Video QoE," in *Proceedings of the 11th International Conference on Quality of Multimedia Experience (QoMEX)*, IEEE, Berlin, Germany, Jun. 2019, pp. 1–6.

[5] K. Borchert, M. Hirth, A. Stellzig-Eisenhauer, and F. Kunz, "Crowd-based Assessment of Deformational Cranial Asymmetries," in *Proceedings of the International Workshop on Crowd-Powered e-Services (CROPS)*, Springer, Trondheim, Norway, Sep. 2019, pp. 145–157.

[6] K. Borchert, M. Hirth, T. Zinner, and A. Göritz, "Designing a Survey Tool for Monitoring Enterprise QoE," in *Proceedings of the 2nd Workshop on QoE-based Analysis and Management of Data Communication Networks (Internet-QoE)*, ACM SIGCOMM, Los Angeles, CA, USA, Aug. 2017, pp. 1–6.

[7] ——, "Collecting Subjective Ratings in Enterprise Environments," in *Proceedings of the 9th International Conference on Quality of Multimedia Experience (QoMEX)*, Springer, Erfurt, Germany, May 2017, pp. 1–2.

[8] K. Borchert, M. Hirth, T. Zinner, and D. C. Mocanu, "Correlating QoE and Technical Parameters of an SAP System in an Enterprise Environment," in *Proceedings of the Fourth IEEE International Workshop on QoE Centric Management (QCMan)*, IEEE, Würzburg, Germany, Sep. 2016, pp. 1–6.

[9] K. Borchert, M. Seufert, K. Hildebrand, and T. Hoßfeld, "QoE Assessment of Enterprise Applications based on Self-motivated Ratings," in *Proceedings of the 12th International Conference on Quality of Multimedia Experience (QoMEX)*, IEEE, Athlone, Ireland, May 2020, pp. 1–6.

[10] K. Borchert, S. Lange, T. Zinner, and M. Hirth, "Identification of Delay Thresholds Representing the Perceived Quality of Enterprise Applications," in *Proceedings of the Second International Workshop on Quality of Experience Management (QoE-Management)*, IEEE, Sardinia, Italy, May 2018, pp. 1–6.

[11] M. Hirth, K. Borchert, F. Allendorf, F. Metzger, and T. Hoßfeld, "Crowd-based Study of Gameplay Impairments and Player Performance in DOTA 2," in *Proceedings of the 4th Workshop on QoE-based Analysis and Management of Data Communication Networks (Internet-QoE)*, ACM, Los Cabos, Mexico, Oct. 2019, pp. 19–24.

[12] M. Hirth, F. Steurer, K. Borchert, and D. Dubiner, "Task Scheduling on Crowdsourcing Platforms for Enabling Completion Time SLAs," in *Proceedings of the 31st International Teletraffic Congress (ITC 31)*, IEEE, Budapest, Hungary, Aug. 2019, pp. 117–118.

[13] K. Borchert, M. Hirth, S. Schnitzer, and C. Rensing, "Impact of Task Recommendation Systems in Crowdsourcing Platforms," in *Proceedings of the Workshop on Responsible Recommendation (FATREC'17)*, ACM, Como, Italy, Aug. 2017, pp. 19–24.

[14] C. Schwartz, K. Borchert, M. Hirth, and P. Tran-Gia, "Modeling Crowdsourcing Platforms to Enable Workforce Dimensioning," in *Proceedings of the International Telecommunication Networks and Applications Conference (ITNAC)*, IEEE, Sydney, Australia, Nov. 2015, pp. 30–37.

[15] M. Becker, K. Borchert, M. Hirth, H. Mewes, A. Hotho, and P. Tran-Gia, "MicroTrails: Comparing Hypotheses about Task Selection on a Crowd Sourcing Platform," in *Proceedings of the 15th International Conference on Knowledge Technologies and Data-driven Business (I-KNOW)*, ACM, Graz, Austria, Oct. 2015, pp. 1–8.

[16] S. Schnitzer, C. Rensing, S. Schmidt, K. Borchert, M. Hirth, and P. Tran-Gia, "Demands on Task Recommendation in Crowdsourcing Platforms - the Worker's Perspective," in *Proceedings of the Workshop on Crowdsourcing and Human Computation for Recommender Systems (CrowdRec)*, ACM, Vienna, Austria, Sep. 2015, pp. 1–6.

Technical Reports

[17] K. Borchert, M. Hirth, M. E. Kummer, U. Laitenberger, O. Slivko, and S. Viete, "Unemployment and Online Labor," ZEW-Centre for European Economic Research, Discussion paper, 18-023, 2018. [Online]. Available: http://ftp.zew.de/pub/zew-docs/dp/dp18023.pdf.

General References

[18] P. Le Callet, S. Möller, A. Perkis, *et al.*, "Qualinet White Paper on Definitions of Quality of Experience," *European Network on Quality of Experience in Multimedia Systems and Services (COST Action IC 1003)*, vol. 3, no. 1.2, 2013.

[19] J. Howe, "The Rise of Crowdsourcing," *Wired Magazine*, vol. 14, no. 6, pp. 1–4, 2006.

[20] E. Estellés-Arolas and F. González-Ladrón-De-Guevara, "Towards an Integrated Crowdsourcing Definition," *Journal of Information Science*, vol. 38, no. 2, pp. 189–200, 2012.

[21] E. Schenk and C. Guittard, "Towards a Characterization of Crowdsourcing Practices," *Journal of Innovation Economics Management*, no. 1, pp. 93–107, 2011.

[22] T. Hoßfeld, M. Hirth, J. Redi, F. Mazza, P. Korshunov, B. Naderi, M. Seufert, B. Gardlo, S. Egger, and C. Keimel, "Best Practices and Recommendations for Crowdsourced QoE-Lessons Learned from the Qualinet Task Force Crowdsourcing," *COST Action IC1003 European Network on Quality of Experience in MultimediaSystems and Services (QUALINET)*, 2014.

[23] U. Gadiraju, R. Kawase, S. Dietze, and G. Demartini, "Understanding Malicious Behavior in Crowdsourcing Platforms: The Case of Online Surveys," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ACM, 2015, pp. 1631–1640.

[24] T. Schulze, S. Seedorf, D. Geiger, N. Kaufmann, and M. Schader, "Exploring Task Properties in Crowdsourcing – An Empirical Study on Mechanical Turk," in *Proceedings of the 19th European Conference on Information Systems (ECIS)*, AIS, 2011, pp. 1–14.

[25] K. Casler, L. Bickel, and E. Hackett, "Separate but Equal? A Comparison of Participants and Data Gathered via Amazon's MTurk, Social Media, and Face-to-face Behavioral Testing," *Computers in Human Behavior*, vol. 29, no. 6, pp. 2156–2160, 2013.

[26] M. Buhrmester, S. Talaifar, and S. Gosling, "An Evaluation of Amazon's Mechanical Turk, its Rapid Rise, and its Effective Use," *Perspectives on Psychological Science*, vol. 13, no. 2, pp. 149–154, 2018.

[27] H. Aris, "Influencing Factors in Mobile Crowdsourcing Participation: A Review of Empirical Studies," in *Proceedings of the 3rd International Conference on Computer Science and Computational Mathematics (ICCSCM)*, 2014, pp. 138–145.

[28] D. Geiger and M. Schader, "Personalized Task Recommendation in Crowdsourcing Information Systems – Current State of the Art," *Decision Support Systems*, vol. 65, pp. 3–16, 2014.

[29] M. Allahbakhsh, B. Benatallah, A. Ignjatovic, H. R. Motahari-Nezhad, E. Bertino, and S. Dustdar, "Quality Control in Crowdsourcing Systems: Issues and Directions," *IEEE Internet Computing*, vol. 17, no. 2, pp. 76–81, 2013.

[30] G. Hsieh and R. Kocielnik, "You Get Who You Pay For: The Impact of Incentives on Participation Bias," in *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, ACM, 2016, pp. 823–835.

[31] W. Mason and D. Watts, "Financial Incentives and the Performance of Crowds," in *Proceedings of the ACM SIGKDD Workshop on Human Computation*, ACM, 2009, pp. 77–85.

[32] N. B. Shah and D. Zhou, "Double or Nothing: Multiplicative Incentive Mechanisms for Crowdsourcing," in *Advances in Neural Information Processing Systems*, 2015, pp. 1–9.

[33] B. McInnis, D. Cosley, C. Nam, and G. Leshed, "Taking a HIT: Designing Around Rejection, Mistrust, Risk, and Workers' Experiences in Amazon Mechanical Turk," in *Proceedings of the CHI Conference on Human Factors in Computing Systems*, ACM, 2016, pp. 2271–2282.

[34] J. Feitosa, D. Joseph, and D. Newman, "Crowdsourcing and Personality Measurement Equivalence: A Warning about Countries Whose Primary Language Is Not English," *Personality and Individual Differences*, vol. 75, pp. 47–52, 2015.

[35] J. Goodman, C. Cryder, and A. Cheema, "Data Collection in a Flat World: The Strengths and Weaknesses of Mechanical Turk Samples," *Journal of Behavioral Decision Making*, vol. 26, no. 3, pp. 213–224, 2013.

[36] U. Gadiraju, J. Yang, and A. Bozzon, "Clarity is a Worthwhile Quality: On the Role of Task Clarity in Microtask Crowdsourcing," in *Proceedings of the 28th Conference on Hypertext and Social Media*, ACM, 2017, pp. 5–14.

[37] M.-H. Wu and A. J. Quinn, "Confusing the Crowd: Task Instruction Quality on Amazon Mechanical Turk," in *Proceedings of the Fifth AAAI Conference on Human Computation and Crowdsourcing*, AAAI, 2017.

[38] J. Rogstadius, V. Kostakos, A. Kittur, B. Smus, J. Laredo, and M. Vukovic, "An Assessment of Intrinsic and Extrinsic Motivation on Task Performance in Crowdsourcing Markets," in *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, AAAI, 2011, pp. 321–328.

[39] J. Cheng, J. Teevan, S. Iqbal, and M. Bernstein, "Break It Down: A Comparison of Macro-and Microtasks," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ACM, 2015, pp. 4061–4064.

[40] A. Papoutsaki, H. Guo, D. Metaxa-Kakavouli, C. Gramazio, J. Rasley, W. Xie, G. Wang, and J. Huang, "Crowdsourcing From Scratch: A Pragmatic Experiment in Data Collection by Novice Requesters," in *Proceedings of*

the Third AAAI Conference on Human Computation and Crowdsourcing, AAAI, 2015, pp. 140–149.

[41] J. Yang, J. Redi, G. Demartini, and A. Bozzon, "Modeling Task Complexity in Crowdsourcing," in *Proceedings of the Fourth AAAI Conference on Human Computation and Crowdsourcing*, AAAI, 2016, pp. 249–258.

[42] S. Komarov, K. Reinecke, and K. Gajos, "Crowdsourcing Performance Evaluations of User Interfaces," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, 2013, pp. 207–216.

[43] R. Khazankin, D. Schall, and S. Dustdar, "Predicting QoS in Scheduled Crowdsourcing," in *Proceedings of the 24th International Conference on Advanced Information Systems Engineering*, Springer, 2012, pp. 460–472.

[44] F. Daniel, P. Kucherbaev, C. Cappiello, B. Benatallah, and M. Allahbakhsh, "Quality Control in Crowdsourcing: A Survey of Quality Attributes, Assessment Techniques, and Assurance Actions," *ACM Computing Surveys (CSUR)*, vol. 51, no. 1, p. 7, 2018.

[45] D. Oleson, A. Sorokin, G. Laughlin, V. Hester, J. Le, and L. Biewald, "Programmatic Gold: Targeted and Scalable Quality Assurance in Crowdsourcing," in *Workshops at the 25th AAAI Conference on Artificial Intelligence*, AAAI, 2011.

[46] A. Checco, J. Bates, and G. Demartini, "All that Glitters is Gold – An Attack Scheme on Gold Questions in Crowdsourcing," in *Proceedings of the Sixth AAAI Conference on Human Computation and Crowdsourcing*, AAAI, 2018.

[47] B. Waggoner and Y. Chen, "Output Agreement Mechanisms and Common Knowledge," in *Proceedings of the Second AAAI Conference on Human Computation and Crowdsourcing*, AAAI, 2014.

[48] S. Jagabathula, L. Subramanian, and A. Venkataraman, "Reputation-based Worker Filtering in Crowdsourcing," in *Advances in Neural Information Processing Systems*, 2014, pp. 2492–2500.

[49] I. Caragiannis, A. Procaccia, and N. Shah, "Modal Ranking: A Uniquely Robust Voting Rule," in *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, AAAI, 2014, pp. 616–622.

[50] M. Bernstein, G. Little, R. Miller, B. Hartmann, M. Ackerman, D. Karger, D. Crowell, and K. Panovich, "Soylent: A Word Processor with a Crowd Inside," in *Proceedings of the 23nd ACM Symposium on User Interface Software and Technology (UIST)*, ACM, 2010, pp. 313–322.

[51] D. Hauser and N. Schwarz, "Attentive Turkers: MTurk Participants Perform Better on Online Attention Checks than Do Subject Pool Participants," *Behavior Research Methods*, vol. 48, no. 1, pp. 400–407, 2016.

[52] U. Gadiraju, P. Siehndel, B. Fetahu, and R. Kawase, "Breaking Bad: Understanding Behavior of Crowd Workers in Categorization Microtasks," in *Proceedings of the 26th ACM Conference on Hypertext and Social Media*, ACM, 2015, pp. 33–38.

[53] S. Rothwell, S. Carter, A. Elshenawy, and D. Braga, "Job Complexity and User Attention in Crowdsourcing Microtasks," in *Proceedings of the Third AAAI Conference on Human Computation and Crowdsourcing*, AAAI, 2016, pp. 20–25.

[54] E. Peer, J. Vosgerau, and A. Acquisti, "Reputation as a Sufficient Condition for Data Quality on Amazon Mechanical Turk," *Behavior Research Methods*, vol. 46, no. 4, pp. 1023–1031, 2014.

[55] T. Schulze, S. Krug, and M. Schader, "Workers' Task Choice in Crowdsourcing and Human Computation Markets," in *Proceedings of the 33rd International Conference on Information Systems (ICIS)*, AIS, 2012, pp. 1–11.

[56] E. Law, C. Dalton, N. Merrill, A. Young, and K. Gajos, "Curio: A Platform for Supporting Mixed-expertise Crowdsourcing," in *Prodeedings of the First AAAI Conference on Human Computation and Crowdsourcing*, AAAI, 2013, pp. 99–100.

[57] A. Kulkarni, M. Can, and B. Hartmann, "Collaboratively Crowdsourcing Workflows with Turkomatic," in *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, ACM, 2012, pp. 1003–1012.

[58] H. Jiang and S. Matsubara, "Efficient Task Decomposition in Crowdsourcing," in *Proceedings of the International Conference on Principles and Practice of Multi-Agent Systems*, Springer, 2014, pp. 65–73.

[59] Y. Tong, L. Chen, Z. Zhou, H. V. Jagadish, L. Shou, and W. Lv, "SLADE: A Smart Large-scale Task Decomposer in Crowdsourcing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 8, pp. 1588–1601, 2018.

[60] T. Hoßfeld, C. Keimel, and C. Timmerer, "Crowdsourcing Quality-of-Experience Assessments," *Computer*, vol. 47, no. 9, pp. 98–102, 2014.

[61] P. Gutheim and B. Hartmann, "Fantasktic: Improving Quality of Results for Novice Crowdsourcing Users," EECS Dept., Univ. California, Berkeley, CA, USA, Tech. Rep. UCB/EECS-2012-112, 2012.

[62] A. Finnerty, P. Kucherbaev, S. Tranquillini, and G. Convertino, "Keep It Simple: Reward and Task Design in Crowdsourcing," in *Proceedings of the Biannual Conference of the Italian Chapter of SIGCHI*, ACM, 2013, pp. 1–14.

[63] V. C. Manam and A. J. Quinn, "Wingit: Efficient Refinement of Unclear Task Instructions," in *Proceedings of the Sixth AAAI Conference on Human Computation and Crowdsourcing*, AAAI, 2018, pp. 108–116.

[64] J. Bragg, D. S. Weld, *et al.*, "Sprout: Crowd-powered Task Design for Crowdsourcing," in *Proceedings of the 31st ACM Symposium on User Interface Software and Technology (UIST)*, ACM, 2018, pp. 165–176.

[65] S. Khanna, A. Ratan, J. Davis, and W. Thies, "Evaluating and Improving the Usability of Mechanical Turk for Low-income Workers in India," in *Proceedings of the First ACM Symposium on Computing for Development*, ACM, 2010, pp. 1–12.

[66] Y. Jiang, J. Kummerfeld, and W. Lasecki, "Understanding Task Design Trade-offs in Crowdsourced Paraphrase Collection," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, 2017, pp. 103–109.

[67] H. Düker and G. Lienert, *Konzentrations-Leistungs-Test: KLT-R.* Göttingen, Germany: Hogrefe, 2001.

[68] E. Starzacher, K. Nubel, and G. Grohmann, *Continous Attention Performance Test.* Göttingen, Germany: Hogrefe, 2006.

[69] D. Dougherty, D. Marsh, and C. Mathias, "Immediate and Delayed Memory Tasks: A Computerized Behavioral Measure of Memory, Attention, and Impulsivity," *Behavior Research Methods, Instruments, & Computers*, vol. 34, no. 3, pp. 391–398, 2002.

[70] R. Brickenkamp, L. Schmidt-Atzert, and D. Liepmann, *Test d2-Revision: Aufmerksamkeits-und Konzentrationstest.* Göttingen, Germany: Hogrefe, 2010.

[71] D. Martin, S. Carpendale, N. Gupta, T. Hoßfeld, B. Naderi, J. Redi, E. Siahaan, and I. Wechsung, "Understanding the Crowd: Ethical and Practical Matters in the Academic Use of Crowdsourcing," in *Evaluation in the Crowd. Crowdsourcing and Human-centered Experiments*, Springer, 2017, pp. 27–69.

[72] R. Brickenkamp, *Test d2: Aufmerksamkeits-Belastungs-Test.* Göttingen, Germany: Hogrefe, 1981.

[73] A. Göritz, "Determinants of the Starting Rate and the Completion Rate in Online Panel Studies," *Online Panel Research: A Data Quality Perspective*, pp. 154–170, 2014.

[74] M. Hirth, "Modeling Crowdsourcing Platforms – A Use-Case Driven Approach," doctoralthesis, Universität Würzburg, 2016.

[75] L. Litman, J. Robinson, and C. Rosenzweig, "The Relationship Between Motivation, Monetary Compensation, and Data Quality Among US- and India-based Workers on Mechanical Turk," *Behavior Research Methods*, vol. 47, no. 2, pp. 519–528, 2015.

[76] M. Eckert, "Slowing down: Age-related neurobiological predictors of processing speed," *Frontiers in Neuroscience*, vol. 5, p. 25, 2011.

[77] K. Kennedy, T. Partridge, and N. Raz, "Age-related Differences in Acquisition of Perceptual-motor Skills: Working Memory as a Mediator," *Aging, Neuropsychology, and Cognition*, vol. 15, no. 2, pp. 165–183, 2008.

[78] B. U. Forstmann, M. Tittgemeyer, E.-J. Wagenmakers, J. Derrfuss, D. Imperati, and S. Brown, "The Speed-accuracy Tradeoff in the Elderly Brain: A Structural Model-based Approach," *Journal of Neuroscience*, vol. 31, no. 47, pp. 17 242–17 249, 2011.

[79] B. Diedenhofen and J. Musch, "cocron: A Web Interface and R Package for the Statistical Comparison of Cronbach's Alpha Coefficients.," *International Journal of Internet Science*, vol. 11, no. 1, 2016.

[80] S. Smith, C. Roster, L. Golden, and G. Albaum, "A Multi-group Analysis of Online Survey Respondent Data Quality: Comparing a Regular USA Consumer Panel to MTurk Samples," *Journal of Business Research*, vol. 69, no. 8, pp. 3139–3148, 2016.

[81] A. Göritz and B. Neumann, "The Longitudinal Effects of Incentives on Response Quantity in Online Panels," *Translational Issues in Psychological Science*, vol. 2, no. 2, p. 163, 2016.

[82] A. Brawley and C. Pury, "Work Experiences on MTurk: Job Satisfaction, Turnover, and Information Sharing," *Computers in Human Behavior*, vol. 54, pp. 531–546, 2016.

[83] R. Brewer, M. R. Morris, and A. M. Piper, "Why Would Anybody Do This?: Understanding Older Adults' Motivations and Challenges in Crowd Work," in *Proceedings of the CHI Conference on Human Factors in Computing Systems*, ACM, 2016, pp. 2246–2257.

[84] B. Winther, M. Riegler, L. Calvet, C. Griwodz, and P. Halvorsen, "Why Design Matters: Crowdsourcing of Complex Tasks," in *Proceedings of the Fourth International Workshop on Crowdsourcing for Multimedia*, ACM, 2015, pp. 27–32.

[85] P.-F. Hsu, H. R. Yen, and J.-C. Chung, "Assessing ERP Post-implementation Success at the Individual Level: Revisiting the Role of Service Quality," *Information & Management*, vol. 52, no. 8, pp. 925–942, 2015.

[86] A. Raake and S. Egger, "Quality and Quality of Experience," in *Quality of Experience: Advanced Concepts, Applications and Methods*, Springer, 2014, ch. 2, pp. 11–33.

[87] U. Reiter, K. Brunnström, K. De Moor, M.-C. Larabi, M. Pereira, A. Pinheiro, J. You, and A. Zgank, "Factors Influencing Quality of Experience," in *Quality of Experience: Advanced Concepts, Applications and Methods*, Springer, 2014, ch. 4, pp. 55–72.

[88] F. Speranza, F. Poulin, R. Renaud, M. Caron, and J. Dupras, "Objective and Subjective Quality Assessment with Expert and Non-expert Viewers," in *Proceedings of the Second International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2010, pp. 46–51.

[89] S. Egger, T. Hoßfeld, R. Schatz, and M. Fiedler, "Waiting Times in Quality of Experience for Web Based Services," in *Proceedings of the Fourth International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2012, pp. 86–96.

[90] ITU-T, "Recommendation P.910," *Subjective Video Quality Assessment Methods for Multimedia Applications*, Apr. 2008.

[91] ——, "Recommendation P.800.2 - Mean Opinion Score Interpretation and Reporting," *Methods for Objective and Subjective Assessment of Speech and Video Quality*, Jul. 2016.

[92] ——, "Recommendation P.808 - Subjective Evaluation of Speech Quality with a Crowdsourcing Approach," *Methods for Objective and Subjective Assessment of Speech and Video Quality*, Jun. 2018.

[93] ——, "Recommendation P.913," *Methods for the Subjective Assessment of Video Quality, Audio Quality and Audiovisual Quality of Internet Video and Distribution Quality Television in any Environment*, Mar. 2016.

[94] ——, "Recommendation P.1501 - Subjective Testing Methodology for Web Browsing," *Methods for Objective and Subjective Assessment of Quality of Services Other Than Voice Services*, Feb. 2014.

[95] T. Zinner, F. Lemmerich, S. Schwarzmann, M. Hirth, P. Karg, and A. Hotho, "Text Categorization for Deriving the Application Quality in Enterprises Using Ticketing Systems," in *Proceedings of the International Conference on Big Data Analytics and Knowledge Discovery*, Springer, 2015, pp. 325–336.

[96] A. Mockus, P. Zhang, and P. L. Li, "Predictors of Customer Perceived Software Quality," in *Proceedings of the 27th International Conference on Software Engineering (ICSE)*, IEEE, 2005, pp. 225–233.

[97] D. Schlosser, B. Staehle, A. Binzenhofer, and B. Boder, "Improving the QoE of Citrix Thin Client Users," in *Proceedings of the IEEE International Conference on Communications (ICC)*, IEEE, 2010, pp. 1–6.

[98] P. Casas, M. Seufert, S. Egger, and R. Schatz, "Quality of Experience in Remote Virtual Desktop Services," in *Proceedings of the IFIP/IEEE International Symposium on Integrated Network Management*, IEEE, 2013, pp. 1352–1357.

[99] W. Bonhag, D. Feindt, S. Olschner, and U. Schubert, "Wie schnell ist "schnell" bei Business-Software? Analyse zur Performance bei der Nutzung von Business-Software," *Mensch und Computer – Usability Professionals*, pp. 22–32, 2015.

[100] M. Dasari, S. Sanadhya, C. Vlachou, K.-H. Kim, and S. R. Das, "Scalable Ground-Truth Annotation for Video QoE Modeling in Enterprise WiFi," in *Proceedings of the IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*, IEEE, 2018, pp. 1–6.

[101] R. Smith and L. A. Kilty, "Crowdsourcing and Gamification of Enterprise Meeting Software Quality," in *Proceedings of the Seventh IEEE/ACM International Conference on Utility and Cloud Computing*, IEEE, 2014, pp. 611–613.

[102] T. Hoßfeld, A. Beyer, A. Hall, A. Schwind, C. Gassner, F. Guillemin, F. Wamser, K. Wascinski, M. Hirth, M. Seufert, *et al.*, "White Paper "Crowdsourced Network and QoE Measurements–Definitions, Use Cases and Challenges"."

[103] W. Van den Broeck, A. Jacobs, and N. Staelens, "Integrating the Everyday-life Context in Subjective Video Quality Experiments," in *Proceedings of the Fourth International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2012, pp. 19–24.

[104] N. Staelens, S. Moens, W. Van den Broeck, I. Marien, B. Vermeulen, P. Lambert, R. Van de Walle, and P. Demeester, "Assessing Quality of Experience of IPTV and Video on Demand Services in Real-life Environments," *IEEE Transactions on Broadcasting*, vol. 56, no. 4, pp. 458–466, 2010.

[105] J. Xue and C. W. Chen, "A Study on Perception of Mobile Video with Surrounding Contextual Influences," in *Proceedings of the Fourth International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2012, pp. 248–253.

[106] D. Guse, S. Egger, A. Raake, and S. Möller, "Web-QoE under Real-world Distractions: Two Test Cases," in *Proceedings of the Sixth International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2014, pp. 220–225.

[107] M. Varela, L. Skorin-Kapov, T. Mäki, and T. Hoßfeld, "QoE in the Web: A Dance of Design and Performance," in *Proceedings of the Seventh International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2015, pp. 1–7.

[108] J.-N. Voigt-Antons, T. Hoßfeld, S. Egger-Lampl, R. Schatz, and S. Möller, "User Experience of Web Browsing-The Relationship of Usability and Quality of Experience," in *Proceedings of the Tenth International Conference on Quality of Multimedia Experience (QoMEX)*, IEEE, 2018, pp. 1–3.

[109] B. Gardlo, M. Ries, T. Hoßfeld, and R. Schatz, "Microworkers vs. Facebook: The Impact of Crowdsourcing Platform Choice on Experimental Results," in *Proceedings of the Fourth International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2012, pp. 35–36.

[110] D. Hands, M. Brotherton, A. Bourret, and D. Bayart, "Subjective Quality Assessment for Objective Quality Model Development," *Electronics Letters*, vol. 41, no. 7, pp. 408–409, 2005.

[111] M. Seufert, J. Kargl, J. Schauer, A. Nüchter, and T. Hoßfeld, "Different Points of View: Impact of 3D Point Cloud Reduction on QoE of Rendered Images," in *Proceedings of the 12th International Conference on Quality of Multimedia Experience (QoMEX)*, IEEE, 2020, pp. 1–6.

[112] P. Hershey, J. Pitts, and R. Ogilvie, "Monitoring Real-time Applications Events in Net-centric Enterprise Systems to Ensure High Quality of Experience," in *Proceedings of the Military Communications Conference (MILCOM)*, IEEE, 2009, pp. 1–7.

[113] W. Bonhag and U. Schubert, "Doppelklicker & Co.–Klickverhalten in Business-Software," *Tagungsband UP09*, pp. 28–32, 2009.

[114] J. Cox, E. Y. Oh, B. Simmons, G. Graham, A. Greenhill, C. Lintott, K. Masters, and J. Woodcock, "Doing Good Online: The Changing Relationships between Motivations, Activity, and Retention among Online Volunteers," *Nonprofit and Voluntary Sector Quarterly*, vol. 47, no. 5, pp. 1031–1056, 2018.

[115] S. Egger-Lampl, J. Redi, T. Hoßfeld, M. Hirth, S. Möller, B. Naderi, C. Keimel, and D. Saupe, "Crowdsourcing Quality of Experience Experiments," in *Evaluation in the Crowd. Crowdsourcing and Human-Centered Experiments*, Springer, 2017, pp. 154–190.

[116] M. Seufert, O. Zach, M. Slanina, and P. Tran-Gia, "Unperturbed Video Streaming QoE Under Web Page Related Context Factors," in *Proceedings of the Ninth International Conference on Quality of Multimedia Experience (QoMEX)*, IEEE, 2017, pp. 1–6.

[117] T. Hoßfeld, M. Seufert, M. Hirth, T. Zinner, P. Tran-Gia, and R. Schatz, "Quantification of YouTube QoE via Crowdsourcing," in *Proceedings of the IEEE International Symposium on Multimedia*, IEEE, 2011, pp. 494–499.

[118] O. Zach, M. Slanina, and M. Seufert, "Investigating the Impact of Advertisement Banners and Clips on Video QoE," in *Proceedings of the 38th International Conference on Distributed Computing Systems (ICDCS)*, IEEE, 2018, pp. 1618–1623.

[119] W. Robitza, P. Kara, M. Martini, and A. Raake, "On the Experimental Biases in User Behavior and QoE Assessment in the Lab," in *Proceedings of the IEEE Globecom Workshops*, IEEE, 2016, pp. 1–6.

[120] J. Persing, H. James, J. Swanson, J. Kattwinkel, C. on Practice, A. Medicine, *et al.*, "Prevention and Management of Positional Skull Deformities in Infants," *Pediatrics*, vol. 112, no. 1, pp. 199–202, 2003.

[121] L. Argenta, "Clinical Classification of Positional Plagiocephaly," *Journal of Craniofacial Surgery*, vol. 15, no. 3, pp. 368–372, 2004.

[122] G. Captier, D. Dessauge, M.-C. Picot, M. Bigorre, C. Gossard, J. El Ammar, and N. Leboucq, "Classification and Pathogenic Models of Unintentional Postural Cranial Deformities in Infants: Plagiocephalies and Brachycephalies," *Journal of Craniofacial Surgery*, vol. 22, no. 1, pp. 33–41, 2011.

[123] J.-F. Wilbrand, "Transferring the Assessment of Cranial Deformities to the Affected," *Journal of Craniofacial Surgery*, vol. 28, no. 2, pp. 303–304, 2017.

[124] B. Ranard, Y. Ha, Z. Meisel, D. Asch, S. Hill, L. Becker, A. Seymour, and R. Merchant, "Crowdsourcing—harnessing the Masses to Advance Health and Medicine, a Systematic Review," *Journal of General Internal Medicine*, vol. 29, no. 1, pp. 187–203, 2014.

[125] J. Tucker, S. Day, W. Tang, and B. Bayus, "Crowdsourcing in Medical Research: Concepts and Applications," *PeerJ - the Journal of Life and Environmental Science*, 2019. [Online]. Available: https://peerj.com/articles/6762/.

[126] S. Ørting, A. Doyle, M. Hirth, A. van Hilten, O. Inel, C. Madan, P. Mavridis, H. Spiers, and V. Cheplygina, "A Survey of Crowdsourcing in Medical Image Analysis," *arXiv preprint arXiv:1902.09159*, 2019.

[127] R. Tse, E. Oh, J. Gruss, R. Hopper, and C. Birgfeld, "Crowdsourcing as a Novel Method to Evaluate Aesthetic Outcomes of Treatment for Unilateral Cleft Lip," *Plastic and Reconstructive Surgery*, vol. 138, no. 4, pp. 864–874, 2016.

[128] P. Meyer-Marcotty, H. Boehm, C. Linz, F. Kunz, N. Keil, A. Stellzig-Eisenhauer, and T. Schweitzer, "Head Orthesis Therapy in Infants with Unilateral Positional Plagiocephaly: An Interdisciplinary Approach to

Broadening the Range of Orthodontic Treatment," *Journal of Orofacial Orthopedics*, vol. 73, no. 2, pp. 151–165, 2012.

[129] L. Motiwalla and J. Thompson, "Enterprise Systems Architecture," in *Enterprise Systems for Management*, Pearson Boston, MA, 2012, pp. 79–109.

[130] D. M. Bahssas, A. M. AlBar, and M. R. Hoque, "Enterprise Resource Planning (ERP) Systems: Design, Trends and Deployment," *The International Technology Management Review*, vol. 5, no. 2, pp. 72–81, 2015.

[131] SAP. (). "SAP Protocol," [Online]. Available: https://help.sap.com/doc/saphelp_nw70ehp1/7.01.16/en-US/4f/992ce8446d11d189700000e8322d00/frameset.htm (visited on 08/01/2020).

[132] D. Strohmeier, M. Mikkola, and A. Raake, "The Importance of Task Completion Times for Modeling Web-QoE of Consecutive Web Page Requests," in *Proceedings of the Fifth International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2013, pp. 38–39.

[133] I. Arapakis, X. Bai, and B. Cambazoglu, "Impact of Response Latency on User Behavior in Web Search," in *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, ACM, 2014, pp. 103–112.

[134] E. Schurman and J. Brutlag. (Jun. 24, 2009). "Performance Related Changes and Their User Impact," Velocity: Web Performance and Operations Conference, [Online]. Available: https://slideplayer.com/slide/1402419/ (visited on 08/01/2020).

[135] J. Brutlag. (Jun. 22, 2009). "Speed Matters for Google Web Search," [Online]. Available: https://services.google.com/fh/files/blogs/google_delayexp.pdf (visited on 08/01/2020).

[136] A. Sackl, P. Casas, R. Schatz, L. Janowski, and R. Irmer, "Quantifying the Impact of Network Bandwidth Fluctuations and Outages on Web QoE," in *Proceedings of the Seventh International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2015, pp. 1–6.

[137] A. Sackl, S. Egger, and R. Schatz, "The Influence of Network Quality Fluctuations on Web QoE," in *Proceedings of the Sixth International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2014, pp. 123–128.

[138] C. Schwartz, T. Hoßfeld, F. Lehrieder, and P. Tran-Gia, "Angry Apps: The Impact of Network Timer Selection on Power Consumption, Signalling Load, and Web QoE," *Journal of Computer Networks and Communications*, vol. 2013, pp. 1–13, 2013.

[139] E. Bocchi, L. De Cicco, M. Mellia, and D. Rossi, "The Web, the Users, and the MOS: Influence of HTTP/2 on User Experience," in *Proceedings of the International Conference on Passive and Active Network Measurement*, Springer, 2017, pp. 47–59.

[140] T. Zimmermann, B. Wolters, and O. Hohlfeld, "A QoE Perspective on HTTP/2 Server Push," in *Proceedings of the Workshop on QoE-based Analysis and Management of Data Communication Networks*, ACM, 2017, pp. 1–6.

[141] H. Z. Jahromi, D. T. Delaney, and A. Hines, "Quantifying the Influence of Browser, OS and Network Delay on Time Instant Metric Measurements for a Web Mapping Application," in *Proceedings of the IEEE 19th International Conference on Communication Technology (ICCT)*, IEEE, 2019, pp. 1580–1584.

[142] A. Saverimoutou, B. Mathieu, and S. Vaton, "A 6-month Analysis of Factors Impacting Web Browsing Quality for QoE Prediction," *Computer Networks*, vol. 164, pp. 1–15, 2019.

[143] H. Jahromi, D. Delaney, and A. Hines, "Beyond First Impressions: Estimating Quality of Experience for Interactive Web Applications," *IEEE Access*, vol. 8, pp. 47 741–47 755, 2020.

[144] D. Strohmeier, S. Jumisko-Pyykkö, and A. Raake, "Toward Task-dependent Evaluation of Web-QoE: Free Exploration vs."Who Ate What?"" In *Proceedings of the IEEE Globecom Workshops*, IEEE, 2012, pp. 1309–1313.

[145] D. Guse, S. Schuck, O. Hohlfeld, A. Raake, and S. Möller, "Subjective Quality of Webpage Loading: The Impact of Delayed and Missing Elements on Quality Ratings and Task Completion Time," in *Proceedings of the Seventh International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2015, pp. 1–6.

[146] M. Varela, T. Mäki, L. Skorin-Kapov, and T. Hoßfeld, "Towards an Understanding of Visual Appeal in Website Design," in *Proceedings of the Fifth International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2013, pp. 70–75.

[147] A. Sackl, K. Masuch, S. Egger, and R. Schatz, "Wireless vs. Wireline Shootout: How User Expectations Influence Quality of Experience," in *Proceedings of the Fourth International Workshop on Quality of Multimedia Experience (QoMEX)*, IEEE, 2012, pp. 148–149.

[148] J. Shaikh, M. Fiedler, P. Paul, S. Egger, and F. Guyard, "Back to Normal? Impact of Temporally Increasing Network Disturbances on QoE," in *Proceedings of the IEEE Globecom Workshops*, IEEE, 2013, pp. 1186–1191.

[149] M. Seufert, S. Wassermann, and P. Casas, "Considering User Behavior in the Quality of Experience Cycle: Towards Proactive QoE-aware Traffic Management," *IEEE Communications Letters*, vol. 23, no. 7, pp. 1145–1148, 2019.

[150] J. Brutlag, Z. Abrams, and P. Meenan. (Mar. 15, 2011). "Above the Fold Time: Measuring Web Page Performance Visually," Web Performance and Operations Conference, [Online]. Available: https://cdn.oreillystatic.com/en/assets/1/event/62/Above%20the%20Fold%20Time_%20Measuring%20Web%20Page%20Performance%20Visually%20Presentation.pdf (visited on 08/01/2020).

[151] P. Reichl, S. Egger, R. Schatz, and A. D'Alconzo, "The Logarithmic Nature of QoE and the Role of the Weber-Fechner Law in QoE Assessment," in *Proceedings of the IEEE International Conference on Communications (ICC)*, IEEE, 2010, pp. 1–5.

[152] S. Egger, P. Reichl, T. Hoßfeld, and R. Schatz, "Time is Bandwidth? Narrowing the Gap Between Subjective Time Perception and Quality of Experience," in *Proceedings of the IEEE International Conference on Communications (ICC)*, IEEE, 2012, pp. 1325–1330.

[153] T. Hoßfeld, F. Metzger, and D. Rossi, "Speed Index: Relating the Industrial Standard for User Perceived Web Performance to Web QoE," in *Proceedings of the Tenth International Conference on Quality of Multimedia Experience (QoMEX)*, IEEE, 2018, pp. 1–6.

[154] T. Tominaga, K. Sato, N. Yoshimura, M. Masuda, H. Aoki, and T. Hayashi, "Web-Browsing QoE Estimation Model," *IEICE Transactions on Communications*, 2017.

[155] A. Balachandran, V. Aggarwal, E. Halepovic, J. Pang, S. Seshan, S. Venkataraman, and H. Yan, "Modeling Web Quality-of-Experience on Cellular Networks," in *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking (MobiCom)*, ACM, 2014, pp. 213–224.

[156] S. Baraković and L. Skorin-Kapov, "Multidimensional Modelling of Quality of Experience for Mobile Web Browsing," *Computers in Human Behavior*, vol. 50, pp. 314–332, 2015.

[157] ——, "Modelling the Relationship between Design/Performance Factors and Perceptual Features Contributing to Quality of Experience for Mobile Web Browsing," *Computers in Human Behavior*, vol. 74, pp. 311–329, 2017.

[158] S. Bashookian. (May 2018). "Machine Learning for Network-Based Prediction of Web Browsing QoE," Politecnico di Torino, [Online]. Available: https://webthesis.biblio.polito.it/8460/1/tesi.pdf (visited on 08/01/2020).

[159] D. N. da Hora, A. S. Asrese, V. Christophides, R. Teixeira, and D. Rossi, "Narrowing the Gap between QoS Metrics and Web QoE Using Above-the-fold Metrics," in *Proceedings of the International Conference on Passive and Active Network Measurement*, Springer, 2018, pp. 31–43.

[160] A. Ben Letaifa, "WBQoEMS: Web Browsing QoE Monitoring System Based on Prediction Algorithms," *International Journal of Communication Systems*, e4007, 2019.

[161] M. Lycett and O. Radwan, "Developing a Quality of Experience (QoE) Model for Web Applications," *Information Systems Journal*, vol. 29, no. 1, pp. 175–199, 2019.

[162] X. Wei, Z. Li, R. Liu, and L. Zhou, "IPTV User's Complaint Prediction based on the Gaussian Mixture Model for Imbalanced Dataset," *Journal of Computers*, vol. 28, no. 6, pp. 216–224, 2017.

[163] L. Wang, J. Jin, R. Huang, X. Wei, and J. Chen, "Unbiased Decision Tree Model for User's QoE in Imbalanced Dataset," in *Proceedings of the International Conference on Cloud Computing Research and Innovations (ICCCRI)*, IEEE, 2016, pp. 114–119.

[164] K. Saeed and A. Muthitacharoen, "To Send or not to Send: An Empirical Assessment of Error Reporting Behavior," *Transactions on Engineering Management*, vol. 55, no. 3, pp. 455–467, 2008.

[165] J. Redi and I. Povoa, "Crowdsourcing for Rating Image Aesthetic Appeal: Better a Paid or a Volunteer Crowd?" In *Proceedings of the International ACM Workshop on Crowdsourcing for Multimedia*, ACM, 2014, pp. 25–30.

[166] A. Ullah, R. B. Baharun, K. Nor, M. Siddique, and A. Sami, "Enterprise Resource Planning (ERP) Systems and User Performance (UP)," *International Journal of Applied Decision Sciences*, pp. 377–390, 2018.

[167] T. Schneider, *SAP Performance Optimization Guide*. Galileo Press, 2009.

[168] Dynatrace and Ixia. (Aug. 1, 2015). "Best Practice Deployment Guide – Dynatrace Data Center RUM using Ixia Network Visibility Solutions," [Online]. Available: https://amasol.de/files/best_practice_deployment_guide_ - _dynatrace _ data _ center _ rum _ using_ - _ixia _ network _ visibility_solutions.pdf (visited on 08/01/2020).

[169] D. Kahneman *et al.*, "Objective Happiness," in Russel Sage, 1999, pp. 3–25.

[170] S. Ickin, K. Wac, M. Fiedler, L. Janowski, J.-H. Hong, and A. K. Dey, "Factors Influencing Quality of Experience of Commonly Used Mobile Applications," *IEEE Communications Magazine*, vol. 50, no. 4, pp. 48–56, 2012.

[171] A. Field, J. Miles, and Z. Field, *Discovering statistics using R*. Sage publications, 2012.

[172] H.-Y. Kim, "Statistical notes for clinical researchers: Chi-squared test and fisher's exact test," *Restorative dentistry & endodontics*, vol. 42, no. 2, pp. 152–155, 2017.

[173] Y. Dodge, "Kolmogorov–smirnov test," in *The concise encyclopedia of statistics*. Springer Science & Business Media, 2008, pp. 283–287.

[174] A. Field, J. Miles, and Z. Field, "Levene test," in *Discovering statistics using R*. Sage publications, 2012, p. 186.

[175] ——, "Welch's independent t-test," in *Discovering statistics using R*. Sage publications, 2012, pp. 368–372.

[176] ——, "Mann-whitney u test," in *Discovering statistics using R*. Sage publications, 2012, p. 921.

[177] ——, "Wilcoxon rank-sum test," in *Discovering statistics using R*. Sage publications, 2012, pp. 655–666.

[178] ——, "Wilcoxon signed rank test," in *Discovering statistics using R*. Sage publications, 2012, pp. 667–673.

[179] ——, "One-way anova," in *Discovering statistics using R*. Sage publications, 2012, pp. 399–412.

[180] ——, "Welch's anova," in *Discovering statistics using R*. Sage publications, 2012, p. 414.

[181] A. C. Tamhane, "Multiple comparisons in model i one-way anova with unequal variances," *Communications in Statistics-Theory and Methods*, vol. 6, no. 1, pp. 15–32, 1977.

[182] A. Field, J. Miles, and Z. Field, "Friedman's test," in *Discovering statistics using R*. Sage publications, 2012, pp. 686–689.

[183] P. Nemenyi, "Distribution-free multiple comparisons," in *Biometrics*, International Biometric Soc 1441 I ST, NW, SUITE 700, WASHINGTON, DC 20005-2210, vol. 18, 1962, p. 263.

[184] A. Field, J. Miles, and Z. Field, "Pearson's correlation coefficient," in *Discovering statistics using R*. Sage publications, 2012, p. 219.

[185] ——, "Point-biserial correlation coefficient," in *Discovering statistics using R*. Sage publications, 2012, p. 229.

[186] ——, "Spearman's correlation coefficient," in *Discovering statistics using R*. Sage publications, 2012, pp. 223–225.

[187] J. Hedderich and L. Sachs, *Angewandte Statistik*. Springer, 2016, p. 112.

ISSN 1432-8801