University
of Basel

Faculty of
Psychology

FAKULTÄT FÜR PSYCHOLOGIE 2003 UNIVERSITÄT BASEL

“The graphic designer was a 5 year old drawing with their toes”: Applying Natural Language Processing to rate user comments of the easyJet Travel app by assessing the User Experience Questionnaire

Master thesis

Wolkow Ewgeni, B.Sc.

Institute of Psychology

Department for Cognitive Psychology and Methodology

University of Basel

November 2019

Thesis Supervisors:

Sharon Steinemann, M.Sc.

Department for Cognitive Psychology and Methodology

Prof. Dr. Klaus Opwis

Department for Cognitive Psychology and Methodology

Abstract

Usability studies and User Experience (UX) surveys are difficult to scale and include problems such as false monetary incentives and low response rates. In addition, global UX assessments rarely reveal why users rate an app in a particular way. The present study attempts to circumvent these problems by applying natural language processing methods to assess an app's UX. 23,498 user comments from the easyJet Travel app were downloaded from Google Play Store and analyzed by applying a multiple linear regression. Regression weights for each semantic differential word of the User Experience Questionnaire (UEQ) were determined and used to create an evaluation on the UEQ dimensions. Additionally, the same evaluation was created for specific app features. The present study was able to extend previous studies by empirically obtaining a rating for semantic differentials of the UEQ based on star ratings provided by real users and evaluate the easyJet Travel app on UEQ dimensions. Lastly, it was possible to apply the UEQ to individual app features.

Acknowledgement

I would like to thank my family who supported me during the last years of my university life. I want to thank them for their love and care and the opportunity they have given me. I would also like to thank my supervisor, Sharon Steinemann, for their expertise and advice over the past year.

Declaration of scientific integrity

The author hereby declares that she/he has read and fully adhered the Code for Good Practice in Research of the University of Basel.

Table of Contents

Introduction

Over the last years, the number of apps that users interact with has been growing constantly. User experience as a field is concerned with the thoughts, feelings and needs of product users. To date, the question of how to efficiently test and assess the user-friendliness of an app still needs answering. The present thesis explores this question by means of natural language processing methods using public user comments of an app which is available in the Google App Store.

In the second half of the 20th century, the psychological research community experienced the so called “cognitive revolution” (Miller, 2003). Whereas hitherto, researchers primarily focused on investigating human behavior, they began to increasingly study cognitive processes. A seminal study from this time is Miller's (1956) paper “The magical number seven, plus or minus two”, that suggested a limit to human processing capacity. It is no coincidence that in 1957, the *Human Factors and Ergonomics Society* was founded, which focuses on achieving “compatibility in the design of interactive systems of people, machines, and environments to ensure their effectiveness, safety, and ease of performance” (*History - the Human Factors and Ergonomics Society*, 2019). Human factors as a field has expanded and introduced new concepts like *usability*, which according to ISO 9241-11 (International Organization for Standardization, 2018), is focused on the effectiveness, efficiency and satisfaction of use. According to Hornbaek (2006), usability can be measured with objective measures (e.g. task completion) or subjective measures like validated questionnaires. From the utilitarian focus of usability on task completion, a more non-utilitarian focus emerged in the early 2000s, which can be subsumed under the term User Experience (UX) (Law et al., 2009).

UX was initially conceptualized focusing on the positive aspects of user interactions, contrasting the focus of usability research, which was preoccupied with the reduction of dissatisfaction by removing usability problems (Hassenzahl et al., 2000). Over the years, the UX research community developed more definitions of UX. Some argued to adopt a more holistic view of user interactions by also focusing on the experiences that followed a specific use situation, even long after it (Law et al., 2009). Others emphasized the importance of repeated measures across the experiential episode since retrospective assessments were not reflective of the whole experience, but rather its most recent episodes (Hassenzahl & Sandweg, 2004). Another view is that UX is a multidimensional construct that focuses, beyond task completion, on symbolic and aesthetic values like e.g. beauty (Hassenzahl, 2004). For a more detailed review of UX concepts see (Bargas-Avila & Hornbæk, 2011).

According to Laugwitz et al. (2008) and in line with the multidimensional view, UX can be conceptualized as partly consisting of pragmatic quality (PQ) and hedonic quality (HQ). PQ is a task-related dimension and means that if a product has high PQ, users can reach task-related goals efficiently and effectively. In contrast, HQ is a non-task related dimension and describes the quality aspects of the user interaction with the product, i.e. aspects like innovativeness or originality. According to this concept, users perceive these two dimensions and average them mentally to obtain a judgement of appealingness (e.g. attractiveness) of the product. The above explanations show that the cognitive revolution in psychology has a long-lasting heritage that has indirectly influenced the emergence of the UX research field.

While usability of products was, and is to this date often tested by observing users interacting with the product, the affective component of user experience is often quantified by means of surveys (Bargas-Avila & Hornbæk, 2011). The benefits of surveys are that they are cost-effective, quick to create, and can be conducted online with tools like Amazons Mturk or FigureEight that enable researchers to access larger groups of people. However, people have to get paid to fill out the surveys, which could also create a false incentive for people to participate for monetary reasons only (Su et al., 2008). Thus, the number of participants and data is often tied to the financial capacity of the researchers. Common problems that researchers using surveys have to face, however, are sampling errors or low response rates (van Selm & Jankowski, 2006).

These problems were partly circumvented in recent years with the rise of natural language processing (NLP) which enabled researchers to access an unprecedented amount of freely available data on the internet.

The aim of this thesis is to combine the research fields of UX and NLP by using validated psychological surveys and textual data. This thesis contributes to the research by assessing the user experience of an app based on empirical data, namely user comments.

Theoretical Background

Natural Language Processing

The language humans use to communicate in their everyday lives is what is referred to as “natural language” in contrast to non-natural languages, such as mathematical notations or programming languages (Bird et al., 2009). Natural language processing is a research field that tries to model and produce human natural language. It is concerned with receiving input (e.g. speech), processing it, e.g. with algorithms, and producing output (e.g. a chatbot answering the user) (Sajnani et al., 2017). This thesis will focus on the processing part of NLP. This subfield can further be distinguished into natural language understanding and grammar parsing. The focus of the former lies in reading text, processing it and understanding the meaning of the text. The latter focuses on determining the syntactical or semantical components of text (Sajnani et al., 2017). NLP is not strictly defined by the methods it uses and can therefore also encompass machine learning and artificial intelligence methods (Cambria & White, 2014).

NLP can be used to analyze data that is voluntarily shared by people on the internet and is freely available to everyone. This data offers the opportunity to run quantitative forms of natural field studies (E. E. Chen & Wojcik, 2016). Such studies are conducted in the natural environment of study subjects and the environment is not manipulated by the researcher (Persaud, 2010).

Psychology and NLP

In psychology, so far, NLP methods have been used tentatively, for instance within family psychology (Atkins et al., 2012) and moral psychology (Hoover et al., 2019; Sagi & Dehghani, 2014). Atkins et al. (2012) used textual transcripts of couple therapy sessions and communication assessments. The dataset contained approximately 6.5 million words. Using topic models and logistic regression models, topics were extracted from the text and behavioral codes like “constructive problem-solving” or “descriptive non-blaming discussions” could be predicted.

Sagi and Dehghani (2014) used 1.8 million New York Times articles to analyze texts in socio-political conflicts like e.g. the World Trade Center attacks in 1993 and 2001. In this example, the results showed that, based on the moral foundation theory (Graham et al., 2013), the moral rhetoric of New York Times journalists changed following the World Trade Center attacks. The moral rhetoric focus shifted towards the dimensions of “harm” and “loyalty”. Other examples where

NLP was used include personality psychology (Park et al., 2015; Plank & Hovy) and clinical psychology (Calvo et al., 2017).

Even though the link between psychology and NLP might not be directly evident, it is more obvious that UX research as a field holds enormous potential in this regard. Since the objects of interest in UX research are often software, it is clear that big data that is accessible on the internet is especially valuable for UX researchers. Data sources that provide researchers with information about the feelings concerning a product and users’ thoughts can potentially be used to evaluate and improve products. Users can use forums, social media and commentary functions to voice their opinion about a product. Smartphone app markets are another instrument used to express opinions. They enable users to download apps, use them, share their opinion after usage, provide helpful feedback to future users or developers and praise or complain about the app by writing a user review and rating the app. One of the biggest app stores, measured by number of available apps, is Google Play Store. It contains roughly 3 million android apps (*Google Play Store: Number of Apps 2019*, 2019). Users can rate an app by giving it a star rating ranging from 1 to 5 stars. According to the website, 1 star stands for “hated it”, i.e. least preferred, and 5 for “loved it”, i.e. most preferred (*EasyJet: Travel App - Apps on Google Play*, 2019). While anyone can get a general idea of the quality of the app by looking at the graphical summary of the star rating (see Figure 1), a graphic does not provide information on why users rated the app the way they did. The user comment (UC) section provides more information in this regard. Users often mention a reason for their rating and at the same time give feedback on how the app could be improved.

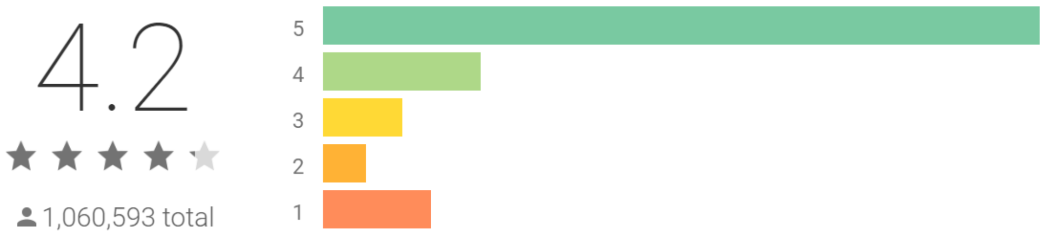

Figure 1. A graphical summary of review ratings as presented in Google Play Store.

Several studies have used comments as a data source to extract meaning from them. Maalej et al. (2016) analyzed about 1.2 million UCs from Apple's and Google's app stores. For each UC they collected title, app name, app category, store, submission date, username, star rating and review text. With this dataset, they used NLP and automatic classification methods to classify each review in the categories "Bug report", "Feature request", "User experience" and "Rating".

Martens and Johann (2017) used over 7 million UCs from Apple store to assess the sentiment of the UCs. They found six different emotional patterns over time (Consistent/Inconsistent Emotion, Sentiment Drops/Jumps, Steady Decrease/Increase). This study shows the invaluable time and cost benefits for automatic assessments. It certainly would have taken the authors much longer to manually evaluate the emotional pattern of 7 million comments. N. Chen et al. (2014) used 173,097 UCs of four apps obtained from Google Play Store. The authors used topic models to classify user comments into "informative" and "non-informative" comments to developers.

Recent studies have tried to extract information about how satisfied or dissatisfied users are with a software (Lima et al., 2017; Yoganathan & Sangaralingam, 2015), the quality of the user comment itself (Ha, 2015) or developed playability heuristics (Zhu et al., 2017).

Although these studies have successfully used NLP methods and dealt with huge amounts of data, their measurements and dimensions lacked a solid theoretical foundation in psychological research.

Previous Work

Automatic UX Evaluation and the User Experience Questionnaire

Rodrigues et al. (2017) conducted multiple studies to assess the sentiment, average star rating and UX of six apps from the Google Play Store. First, ML classification methods were used to find out if the star ratings could be predicted based on the content of the UC. The results suggested a weak relationship between the two. Next, the authors manually assessed the sentiment of every UC based on the content of the UC and assigned a value from 1 (“hating it”) to 5 (“loving it”), which took them five and a half weeks to do. The UX of every app was assessed using an expert-based evaluation, i.e. a heuristic evaluation. This evaluation was manually conducted using UX guidelines during a period of seven days. The authors observed a positive relationship between the review sentiment and the UX evaluation. Lastly, ML methods were used to predict the sentiment of reviews from its contents. The results suggested that ML can better infer the sentiment from the UC contents than the star rating.

Although these studies have used ML methods and the UC content to infer the UC sentiment and star rating, it is not surprising that the sentiments were better predicted than the star rating. The sentiments were subjectively assigned by the study authors and were based on the words used in the comments. Users might assign a different value to the words they use, e.g. someone could rate an app with three stars and still add a comment “loved it”. Further, the UX evaluation was not only time-consuming, but also highly subjective since it was based on only the evaluations of a handful of experts who are always influenced by their previous knowledge. Issues that might occur after a longer usage time than seven days might not have been detected.

Aciar and Aciar used a more standardized approach in their 2017 study by combining NLP methods with the User Experience Questionnaire (UEQ), which is a validated measurement of User Experience (Laugwitz et al., 2008) and will be briefly explained in the following.

The UEQ consists of 26 semantic differentials that measure the pragmatic and hedonic quality of an app on six dimensions, namely Attractiveness, Perspicuity, Efficiency, Dependability, Stimulation and Novelty. Semantic differentials are a method of measuring associations between concepts (e.g. lady) and a pair of polar terms (e.g. rough-smooth). It can be used with a 7-point Likert scale and measures the direction as well as the intensity of the association (Osgood, 1952). The UEQ is supposed to rapidly and directly measure UX (Schrepp et al., 2017) by allowing users to express the attitudes, feelings and impressions that surfaced in the interaction with the product. The first dimension Attractiveness is conceptualized as the pure valence of the user experience and is therefore dependent on the other dimensions. The dimensions Perspicuity, Efficiency and Dependability measure the pragmatic quality of the app, i.e. the usability. The last two dimensions, Novelty and Stimulation, measure the hedonic quality of the app (Schrepp et al., 2017). The Attractiveness scale consists of six and all other dimensions of four semantic differentials. Additionally, it is a sufficiently reliable instrument (Cronbach’s alpha between .69 and .88) with a good construct validity (Laugwitz et al., 2008).

Aciar and Aciar (2017) created a rule-based algorithm and used textual hotel reviews for 100 hotels on TripAdvisor to first assess a UC’s sentiment and then assign a numerical value (either positive or negative) to each UEQ dimension, which represents the UX of the app. The authors split each review into sentences and each sentence into words. The words, referred to as tokens, were the analyzed elements. For instance, the sentence “This hotel is secure and friendly” would have been split into the tokens “this”, “hotel”, “is”, “secure”, “and” and “friendly”. To check which words were relevant for the UX, the authors created a user-defined dictionary (UDD) that was based on the UEQ semantic differentials. The words in this UDD were assigned a fixed value. For example, the words in the differential “secure – not secure” would receive a rating of +1 for “secure” and -1 for “not secure”. In the example sentence “the hotel is secure and friendly” the algorithm would have checked if the tokens occurred in the UDD and would summarize and calculate the mean of all occurrences for each sentence. The resulting value in this example would be +1, because “secure” and “friendly” both occur in the UDD with a value of +1 and the mean of these two words would also be +1. As a result, each hotel in Aciar and Aciar’s study had a rating with one value on every UEQ dimension. In theory, if the UEQ had been applied as a survey version, the hotel ratings would have the same result structure, i.e. six values, one for each dimension. However, Aciar and Aciar (2017) were able to provide this rating to 100 hotels simultaneously. Again, this shows the enormous time and cost savings through the use of NLP methods in UX research.

Lechler and Burghardt (2017) used online survey data on users’ current use of smartphones. In the survey, users were additionally asked to complete an online version of the UEQ survey regarding their user experience with the smartphone so far. The authors analyzed the textual survey data with their self-created NLP tool “UxMiner”. The tool detected words of the semantic differentials of the UEQ (UEQ words) in the text and gave them a default rating of 4 (based on the UEQ Likert-scale

1 to 7). For every additional word occurrence, the rating increased by 1 point. Further, the tool included rules to increase and decrease the rating based on comparatives, superlatives, attenuators and negation, while never exceeding or dropping below the scale range of 1 and 7. Using the UxMiner the authors were able to rate two smartphone products on all six UEQ dimensions and compared the UxMiner evaluation with the results of the survey version of the UEQ. The results showed that the two ratings differed and therefore the UxMiner could not reliably assess the UEQ dimensions compared to the UEQ as the gold standard.

Aim of the Study

Despite the benefits in the methods applied in previous works, the approaches also had its downsides. It is not self-evident that people do always agree on the meaning of words and how strongly these words represent their feelings or opinions. Some people may use different words to express the same intensity of a feeling, or have different feelings but still use the same word, which may have led raters in the study of Rodrigues et al. (2017) to rate the sentiment of UCs differently than users perceived it. The approaches of Aciar and Aciar (2017) and Lechler and Burghardt (2017) forced a pre-defined value upon words, which may correspond with the way how users perceived the words. To address this problem, the present thesis proposes a procedure to determine the polarity of words based on the star rating of the UC and use this to rate the UX of an app. In the literature (Langhe et al., 2016) star ratings have been understood as an indicator for product quality.

As shown in the theoretical background, UX is a multidimensional construct that can be understood as the users’ perception of pragmatic and hedonic product qualities, therefore it seems reasonable to use star ratings as an approximation of user experience. It remains unclear to which extent the UEQ can be recreated with NLP methods. This thesis tries to address this problem with research question 1 (RQ1).

> RQ1: To which extent is it possible to assess the dimensions of the UEQ with NLP?

Aciar and Aciar (2017) as well as Rodrigues et al. (2017) and Lechler and Burghardt (2017) focused on app-level assessments, but none of them focused on which app features exactly caused good or bad UX. Therefore, their evaluations were only applied to an app as a whole. One challenge in UX research is to understand why users experience an app the way they do. This is relevant for the improvement of app development and for other users as well, thus research question 2 is focused on answering this.

> RQ2: To which extent is it possible to rate app features with NLP based on the UEQ?

Methods

In general, big data research projects can be split into separate phases, namely data management planning, acquisition of data, pre-processing of data and data analysis (E. E. Chen & Wojcik, 2016). In the following, in each subsection, the focus will be on one of these steps.

Data Management Planning

This thesis aims to analyze user comments and rate an app on a validated UX scale (UEQ) on the one hand, and on the other hand, associate app features with each UX dimension. In the planning phase it is crucial to estimate the amount of data that will be needed, since it has to be stored, analyzed and maintained.

To determine how many comments are needed for the analysis, the datasets of Aciar and Aciar (2017), Lechler and Burghardt (2017) and Rodrigues et al. (2017) were compared. Their datasets had between 1,000 (Aciar & Aciar, 2017) and 4,500 (Rodrigues et al., 2017) user comments per app. Based on this information, I planned to create a dataset of at least 4,500 user comments.

The dataset containing the user comments was stored in a csv file to enable a simple input in the statistics program R. To ensure anonymity, usernames were omitted from the dataset and UCs were numbered.

The basis for the analysis consisted of three datasets, the first one containing the user comments, the second one the list of app features and the last one the list of UEQ words.

Data Acquisition

To extract the needed comments for the analysis, Google Play Store was used. This store was chosen because it is accessible without an account and is one of the biggest app stores (*Infographic: The Biggest App Stores*, 2019). The next four subsections describe how the data was acquired and describe each dataset.

Choosing an app. To ensure that the RQs were properly answered, the app of which comments were examined had to meet certain criteria. First, the app needed to be feature-rich, so multiple features could possibly be detected. Additionally, these features needed to have clear names, since the algorithm detects them by their written forms. Second, the app needed to have enough comments, i.e. more than 4,500 UCs, as mentioned above. With these constraints, the Google Play Store was searched for apps whilst reading the app description. The app “easyJet: Travel app” (in the following referred to as easyjet app) was chosen because it had a reasonable amount of comments (23,498 comments) and included features which could be named by distinct names, e.g. “scan”, “boarding pass” or “flight tracker”. For the full list of features see Table 2. The English version of the easyjet app was used since the study was conducted in English and this version contained more UCs.

User comments. A common way to download data from a webpage is to scrape it. Scraping means to automatically read the HTML code of the webpage and download targeted HTML tags. To achieve this task, I developed a custom webscraper in Python (see Appendix A). The username, date, comment text and star rating of all comments were scraped and saved as a csv file. The csv file can be found in Appendix B. In total, the dataset consisted of 23,498 comments which all had star ratings. Table 1 presents the frequency of the star ratings based on user comments.

Table 1

Frequency of different ratings, counted by user comments

	Star ratings				
	1 star	2 stars	3 stars	4 stars	5 stars
Frequency	2004	711	939	4466	15377

App features. The description text of the easyjet app was analyzed and app features that could be used as keywords were used to create an app feature list. Developers have the option to describe the functionalities of their app in this description. In addition to this description text analysis, the app was downloaded, app functionalities were explored and functions with explicit names were added to the app feature list. This resulted in a list of 27 app features which is shown in Table 2.

Table 2

App features found in the app description and the app (original words), words that were consolidated and the stemmed version of all app feature words

Original words	Consolidated words	Stemmed version
book		book
scan		scan
cards		card
camera		camera
manage		manag
view		view
change flight	change_flight	change_flight
seats		seat
bags		bag
sport equipment	sport_equipment	sport_equip
check in	check_in	check_in
passport		passport
boarding pass	boarding_pass	boarding_pass
store		store
offline		offlin
data connection	data_connection	data_connect
flight tracker	filght_tracker	filght_track
location		locat
real time	real_time	real_tim
arrival		arriv
departure		departur
live updates	live_updates	live_upd
mobile host	mobile_host	mobile_host
reminder		remind
gate		gate
baggage reclaim	baggage_reclaim	baggage_reclaim
belt		belt

UEQ words. The English version of the UEQ was downloaded from www.UEQ-online.org [accessed 19.09.2019] (Laugwitz et al., 2008) and was used to create a list of UEQ words. As a next step, a dataset was created which included all semantic differentials of every dimension of the UEQ. Each semantic differential was split into positive and negative dimension words and can be seen in column 1 of Table 3. This ensured that a separate linear regression weight for each positive and negative UEQ word could be calculated.

All three of the above-mentioned datasets needed to be compatible, i.e. the same words needed to be written in the same way. This was achieved in the data pre-processing stage.

Table 3

Original, replaced and stemmed versions of UEQ along with words that were consolidated under the stemmed version

Original UEQ words	Manually replaced UEQ words	Stemmed version of UEQ words	Words consolidated under the stemmed version
enjoyable	-	enjoy	enjoy
good	-	good	good, goodbye, goodies, goodnight, goods
pleasing	-	pleas	please, pleased, pleasing
pleasant	-	pleasant	pleasant, pleasantly
attractive	-	attract	attractive
friendly	-	friendli	friendliness
annoying	-	annoi	annoy, annoyed, annoying, annoys
bad	-	bad	bad
unlikable	-	unlik	unlike
unpleasant	-	unpleas	unpleasant
unattractive	-	unattract	-

Table 3 (continued)

unfriendly	-	unfriendli	unfriendly
fast	-	fast	fast, faste
efficient	-	effici	efficiency, efficient, efficiently
practical	-	practic	practical, practice, practicing
organized	-	organ	organize, organized
slow	-	slow	slow, slowing
inefficient	-	ineffici	inefficient
impractical	-	impract	-
cluttered	-	clutter	clutter, cluttered
understandable	-	understand	understand, understanding
easy to learn	simple	simpl	simple
easy	-	easi	easi, easiness, easy
clear	-	clear	clear, cleared
not understandable	unclear	unclear	unclear
difficult to learn	difficult	difficult	difficult
complicated	-	complic	complicated, complications
confusing	-	confus	confuse, confused, confusing, confusion
predictable	-	predict	predictable, prediction, predictive
supportive	-	support	support, supported, supporting, supports
secure	-	secur	secure, secured
meets expectation	expected	expect	expect, expectations, expected
unpredictable	-	unpredict	-
obstructive	-	obstruct	-
not secure	unreliable	unreli	unreliable
does not meet expectation	disappointing	disappoint	disappointed, disappointing, disappointment
valuable	-	valuabl	valuable
exciting	-	excit	excited, excitement, exciting
interesting	-	interest	interest, interested, interesting, interests
motivating	-	motiv	-
inferior	-	inferior	inferior

Table 3 (contined)

boring	-	bore	bore, boring
not interesting	tedious	tediou	tedious
demotivating	-	demotiv	-
creative	-	creativ	-
inventive	-	invent	invention
leading edge	forefront	forefront	-
innovative	-	innov	innovation, innovative
dull	-	dull	dull
conventional	-	convent	-
usual	-	usual	usual, usually
conservative	-	conserv	-

Data Pre-Processing

The success of big data methods is strongly dependent on the quality of the data. In order to ensure the replicability of the study, the data pre-processing must also be described precisely, since the results can change depending on the data pre-processing. Before the analysis, data needs to be pre-processed by removing unwanted stopwords, stemming, transforming words to lowercase and tokenization.

Stopwords are words that often serve a helping function in a sentence (e.g. “the”, “or”, “I”, “to”) and do not carry important information for the task at hand. Also, these words occur very often in natural language and can thus obstruct the algorithm from finding meaningful results. Stemming is the procedure of reducing words to their word stem (Maalej et al., 2016). This way, words like “argue”, “arguing”, “argued” are reduced to their word stem “argu”. This step groups words with identical word stems together and expands the word counts for words with similar meaning. Tokenization is a process of delimiting a string into subsections, based on specific rules. The sentence “The quick brown fox jumps over the lazy dog” could e.g. be tokenized by using a space (“ “) as a delimiter, which would result in 9 tokens (one for each word). The full pre-processing procedure to obtain a dataset ready to be analyzed is explained in the following steps 1 to 11. Figure 2 summarizes the data acquisition and data pre-processing steps.

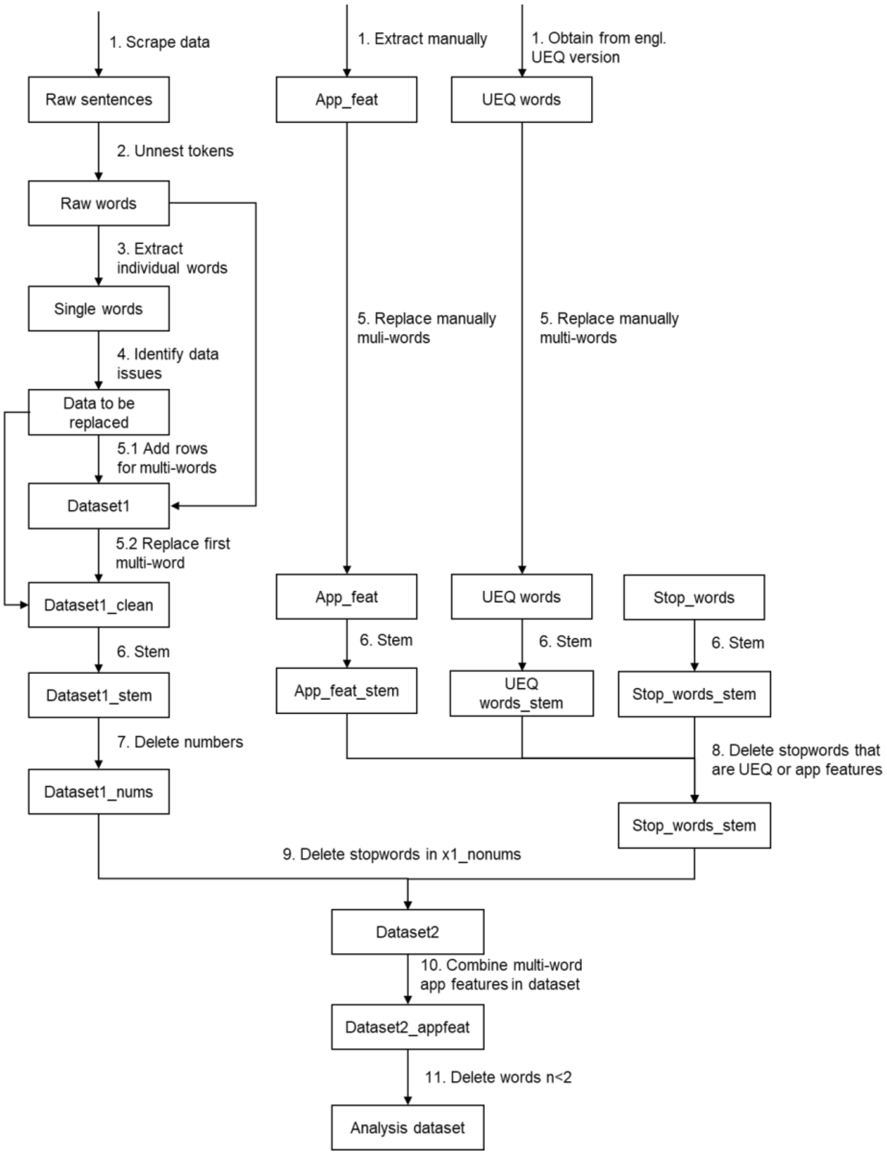

Figure 2. Data Acquisition and Data Pre-Processing steps.

Step 1. The data acquisition was explained in the previous section and is only mentioned here for the sake of completeness.

Step 2. Based on the spaces between words, sentences are split and tokenized which resulted in 277,253 tokens.

Step 3. Taking the raw word dataset, individual words were extracted. This resulted in 8,908 individual words.

Step 4. This unique word dataset was manually analyzed for data quality issues and word forms that would not be detectable by the stemming algorithm. Words that needed to be corrected were added to a separate list. Words either needed to be standardised (e.g. recognise – recognize), corrected (e.g. aces to access) or split into individual words (e.g. “flight.details” to flight and details). 1,048 words were detected that needed replacing. Issues found were wrong spelling (e.g. “bagage” instead of baggage), typos (e.g. “because” instead of because), slang (e.g. ‘hussle’ instead of hustle), merged words (e.g. „afortenight“), or numbers and dates (e.g. 04. Apr). Words with apostrophes were replaced by words without apostrophes (e.g. easy’s by easy) because the stemming algorithm is not able to stem such cases correctly.

Step 5. The app feature list and the UEQ word list contained items that consisted of multiple words like “boarding pass” or “meets expectation”. To ensure a clear detection of app features in the subsequent analysis, these multiple words were combined by an underscore, i.e. in this case boarding_pass (see column 2 of Table 2). The semantic differentials in the UEQ consisting of more than one word (multi-words) were not combined by an underscore but replaced by a single word (see column 2 of Table 3). The reason is that the probability of finding a single word in the dataset is higher than finding a specific combination of words. Additionally, two words can often be separated by other words, e.g. “meets my expectation”, which would add additional complexity to the analysis. Hence, to avoid this, synonyms were used to replace them. Eight word combinations were identified and replaced by synonyms (e.g. “leading edge” replaced by “forefront”). The replacements can be found in column 2 of Table 3. The replacements were checked with two English native speakers to ensure that they cover the multi-word versions’ meaning.

Step 5.1. The resulting list from Step 4 is used to create one new row for every word that will be split into two words (e.g. “he’ll) and inserts the second word (e.g. “will”). This example would have “he’ll” in one row and “will” in the following row.

Step 5.2. Next, words that initially needed splitting were replaced by the first word they include. To finish the example from the last step, “he’ll” would now be replaced by “he” and the following row would now have the entry “will”. As a result of steps 5.1 and 5.2 every word in the dataset would now be an unabbreviated version.

Step 6. The app feature list, the stopword list, the UEQ list and the UC dataset were stemmed. This means that words with the same meaning were automatically consolidated under the same stem. The stemmed UEQ words can be found in column 4 of Table 3 and the stemmed app feature words in column 3 of Table 2.

Step 7. Numbers that remained in the dataset were manually deleted.

Step 8. The stopword list was compared with the UEQ and app feature list. If either an app feature or an UEQ word was present in the stopword list, it was deleted from the stopword list.

Step 9. Using the “cleaned up” version of the stopword list, all stopwords in the UC dataset were deleted. This resulted in 122,106 remaining words.

Step 10. Since some app features were combined by an underscore in the app feature list, the same words were now combined in the same manner in the UC dataset to ensure that they could be found by the algorithm.

Step 11. In the last step, words that occur only once in the whole UC dataset were deleted to shorten the regression calculation time. This resulted in the final analysis dataset, which was the basis for the data analysis. This dataset consisted of 115,101 words and 2,555 unique words.

Data analysis

The following section is split into two parts. The first subsection explains how RQ1 was answered by calculating a multiple linear regression (MLR) on the comment's star rating using the UC words as independent variables. The second subsection explains how RQ2 was answered by determining co-occurrences for each UEQ word with app features and using a self-developed algorithm to rate app features in the UEQ dimensions.

Research Question 1. Rather than assigning a fixed value to each word, as in Aciar and Aciar's (2017) study, I propose to use the weights of a MLR as a rating for each word. Usually linear regressions are applied to interval scaled independent variables. In the case of the nominal scaled independent variable, as in the present study, the weights represent simple averages of the star ratings. Thus, each word was assigned the average star rating of the sentences in which it appeared.

Since the semantic differentials of the UEQ were split into positive and negative words, each UEQ dimension was also split into positive and negative subdimensions as well. This resulted in six positive and six negative subdimensions. Each UEQ word weight will be averaged per subdimension and both subdimensions again averaged per corresponding dimension using Equation (1). This will result in six values, one for each UEQ dimension, equivalent to the results of the survey version of the UEQ.

Figure 3 shows the step by step analysis for RQ1. In the following, each step will be explained to obtain an app rating equivalent to the survey UEQ rating.

$$R_d = \frac{\frac{\sum_p (x_{pd} * N_{pd})}{\sum_p N_{pd}} + \frac{\sum_j (x_{nd} * N_{nd})}{\sum_n N_{nd}}}{N_{pd} + N_{nd}} \tag{1}$$

p = Index for positive subdimension words

n = Index for negative subdimension words

d = One of the UEQ dimensions

R_d = Evaluation value of dimension d

N_d = Number of words in dimension d

x_d = UEQ word in dimension d

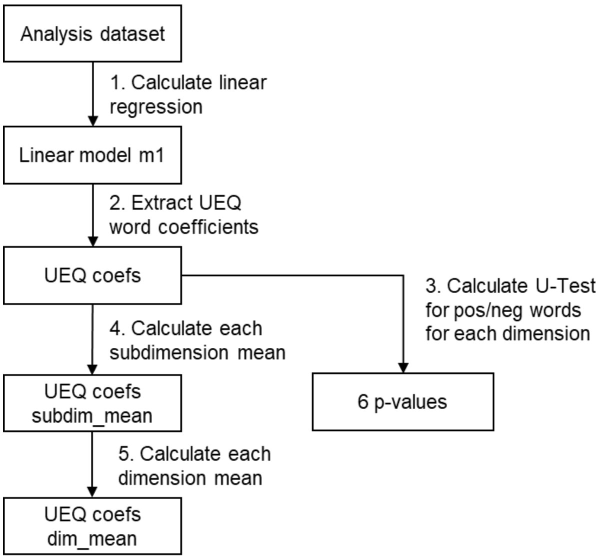
Analysis dataset
1. Calculate linear regression
Linear model m1
2. Extract UEQ word coefficients
UEQ coefs
3. Calculate U-Test for pos/neg words for each dimension
4. Calculate each subdimension mean
UEQ coefs subdim_mean
6 p-values
5. Calculate each dimension mean
UEQ coefs dim_mean

Figure 3. Process of an UEQ evaluation based on UEQ words.

Step 1. The MLR model was calculated with star ratings as the dependent variable and UC words as the independent variable. This resulted in 2,555 regression weights.

Step 2. The regression weights for each UEQ word were extracted. (for table of weights, see Table 6.)

Step 3. For each UEQ dimension individually, the weights were tested with a directed Mann-Whitney-U-Test (U-test) for differences. The alternative hypothesis was that the positive weights were lower than the negative weights. Hence, the test was applied six times, once for each UEQ dimension.

Step 4. The weighted average was taken per UEQ subdimension, resulting in six positive and six negative average weights (for subdimension means see Table 7).

Step 5. The positive and negative subdimension means were averaged for each UEQ dimensions, resulting in six values, one for each dimension. These values are then converted to a scale with a minimum of - 3 and a maximum of + 3 using Equation (2). This creates a rating on the same scale as the survey UEQ. This rating is presented on the same scale as the result of a survey UEQ.

$$x_2 = \frac{(max_{new} - min_{new}) * (x_1 - min_{old})}{(max_{old} - min_{old})} + 1 \qquad (2)$$

x_2 = New value on a - 3 to + 3 scale

x_1 = Original value on a 1 to 5 scale

max_{new} = + 3

min_{new} = - 3

max_{old} = 5

min_{old} = 1

Research Question 2. Co-occurrences of UEQ words and app features in the same sentence offer an option to assign app features to the corresponding UEQ dimensions.

Simple co-occurrence might not be enough to extract valuable insights for how app features should be rated because app features might appear in one sentence with the positive UEQ word and in another with the corresponding negative UEQ word. Here, the number of occurrences might offer a way to interpret the findings and will be presented in the result section as well. To use the co-occurrences and total occurrences information for assigning and rating the app feature for one semantic differential, I used Equation (3).

$$r_x = \frac{\left(w_P \cdot N_p\right) + \left(w_n \cdot N_n\right)}{N_p + N_n} \tag{3}$$

w_P = Weight of positive UEQ word

w_n = Weight of negative UEQ word

N_p = Number of occurrences with the positive UEQ word

N_n = Number of occurrences with the negative UEQ word

r_x = Rating of word x for the corresponding semantic differential

Equation (3) is applied when an app feature co-occurred at least with the positive or negative word part of the semantic differential. If the app feature only occurred with one part of the semantic differential, the rating will automatically result in the linear regression weight of the co-occurred positive/negative UEQ word. In cases where the app feature was not associated with either the positive or negative semantic differential, r_x will not be calculated.

To calculate the overall rating for a specific app feature (e.g. “boarding pass”) in a UEQ dimension (e.g. Efficiency) all r_x’s will be averaged and result in one value for each UEQ dimension. The process steps to obtain one value for each UEQ dimension for each app feature can be seen in Figure 4 and will be explained in the following.

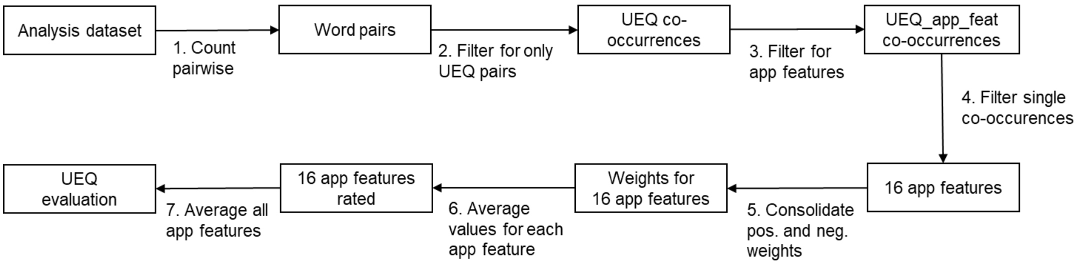
Analysis dataset
1. Count pairwise
Word pairs
2. Filter for only UEQ pairs
UEQ co-occurrences
3. Filter for app features
UEQ_app_feat co-occurrences
4. Filter single co-occurences
16 app features
5. Consolidate pos. and neg. weights
Weights for 16 app features
6. Average values for each app feature
16 app features rated
7. Average all app features
UEQ evaluation

Figure 4. Process of obtaining app feature ratings and an UEQ evaluation.

Step 1. All co-occurrences in the analysis dataset were obtained.

Step 2. These co-occurrences were filtered to only show the co-occurrences of UEQ words.

Step 3. Then these co-occurrences were filtered to only show co-occurrences of UEQ words with app features.

Step 4. To find out how many app features were mentioned, the dataset was filtered to show only unique app features.

Step 5. Equation (3) was applied to each semantic differential of every app feature.

Step 6. For each app feature all semantic differentials were averaged to one value per dimension.

Results

The results section will be split into three parts. In the first part will, results of general app feature word and UEQ word occurrences in the dataset will be presented. This is important for answering RQ1 and RQ2. In the remaining parts, results to RQ1 and RQ2 will presented. All results will be presented with the stemmed word versions.

General occurrences

The frequency of UEQ words and app feature words found in the dataset can be seen in Table 4. Thirteen of the 52 UEQ words could not be found in the dataset. The following words did not appear in any user comment: enjoy, unpleas, unattract, creativ, forefront, dull, convent, conserv, motiv, demotiv, unpredict, obstruct and impract. Comparing these words with column 4 of Table 2, it becomes clear that although “enjoy” and “unpleasant” were in the original unstemmed dataset, they were not found in the pre-processed dataset. The reason is that “enjoyable” was stemmed to “enjoy”, but in the UC dataset “enjoy” was stemmed to “enjoi”, which led to the mismatch. “Unpleasant” occurred only once and was therefore deleted in the pre-processing. As seen in Table 4, roughly 80% of the total UEQ word occurrences consisted of only three words, namely “easi” (54%), “good” (16.78%) and “simpl” (9.14%). The rest, i.e. 39 UEQ words occurred in 12,579 UCs (53.54% of all UCs).

26 out of 27 app features were found in the dataset, which occurred in 6,238 UCs (26.55% of all UCs). The only app feature not mentioned was mobile_host. Apparently, users tend to speak mainly about three app features (book, boarding_pass and check_in). This is important since with RQ2 I try to combine app

feature words with UEQ words and the probability of doing this is higher, the more frequently a word occurs in the dataset. In other words, the more often users mention app features or use UEQ words in their comments, the higher is the probability that these words co-occurr. Also noteworthy is the fact that overall, UEQ words occurred more often than app feature words, Which might be explained by the fact that app features were described with more than one UEQ word as can be seen in the UC: “very quick clear and easy to book flights on this app”. However, users also used UEQ words to describe other aspects than app features as seen in the UC: “The app won't load. Cleared my playstore data and cache.... nothing”. In this UC, the word “cleared” was stemmed to “clear” although it meant deleting something, instead of having a clear understanding of something.

Table 4.

Occurrences of UEQ words and app feature words in the analysis dataset.

App feature word	Occurrence	UEQ dimension	UEQ word	Occurrence
book	3357	Attractiveness	easi	8543
boarding_pass	2382	Attractiveness	good	2652
check_in	1599	Perspicuity	simpl	1444
card	379	Efficiency	fast	625
gate	283	Attractiveness	friendli	415
view	277	Perspicuity	clear	401
manag	214	Attractiveness	pleas	271
seat	207	Efficiency	effici	255
passport	200	Attractiveness	bad	139
scan	188	Efficiency	slow	112
departur	111	Attractiveness	annoi	106
store	111	Dependability	expect	102
bag	101	Perspicuity	understand	101
change_flight	82	Dependability	secur	91

Table 4 (contined)

arriv	73	Efficiency	practic	81
offlin	63	Dependability	disappoint	66
remind	62	Dependability	support	53
filght_tracker	60	Perspicuity	difficult	51
camera	43	Novelty	usual	49
live_updates	25	Perspicuity	confus	48
real_time	24	Attractiveness	unlik	35
locat	24	Perspicuity	complic	28
belt	15	Dependability	unreli	23
data_connection	9	Attractiveness	pleasant	19
sport_equipment	3	Efficiency	clutter	14
baggage_reclaim	2	Stimulation	interest	14
		Efficiency	organ	9
		Stimulation	bore	7
		Attractiveness	attract	6
		Stimulation	excit	6
		Perspicuity	unclear	5
		Novelty	innov	5
		Stimulation	tediou	4
		Attractiveness	unfriendli	4
		Stimulation	valuabl	3
		Dependability	predict	3
		Stimulation	inferior	3
		Novelty	invent	2
		Efficiency	ineffici	2
Total	9894		Total	15800

Note. UEQ words and app feature words that did not occur in the dataset are not shown here.

Results for RQ 1

RQ1 examines the extent to which it is possible to assess the dimensions of the UEQ with NLP.

Weights for UEQ words. The multiple linear regression was calculated on the whole dataset for each word (n=115,101 words), which resulted in 2,555 regression weights for each individual word. For 39 out of 52 UEQ words, linear regression weights were found in the analysis dataset and are presented in Table 6. Of these findings, 32 were original UEQ words and 7 initially manually replaced words. Of the 13 missing values, 12 were original UEQ words and one was an initially manually replaced word. On the one hand this could suggest that the initial replacement of multi-words was mostly successful, but on the other hand this does not mean that the replaced words were capturing the same meaning as the multi-words. This is seen in the UC quote: “Hopeless unreliable website but great app”, which was rated with 5 stars and therefore the UEQ word “unreliable” was given the same star rating, although it should be in theory rated with fewer stars. This example also shows that although “unreliable” and “app” were in the same UC comment, they did not relate to each other. This presents a shortcoming of the used method.

Figure 4 presents the found weights for each UEQ word, which correspond with the star rating they predict, sorted by the corresponding UEQ dimension and subdimension. The weights ranged from 1 to 4.89 stars (*M* = 3.56, *SD* = 1.08). Further, Figure 4 shows only UEQ words that were found and omits UEQ words that were not present in the analysis dataset.

Contrary to expectations, some negative words (“complic”, “unlik”, “usual”) had high weights which can partly be explained by negations and attenuators (e.g.

"more" or "less") as seen in the UC quote: "It's easy and makes the travelling less complicated. You don't have to worry about printing boarding passes or if forgotten" and "Perfect app with no complications, easy to use". Moreover, they were used with different semantic meanings: "Unlike other budget airline apps, this app is very professional and does all the basics required. Very good", which can be attributed to the stemming.

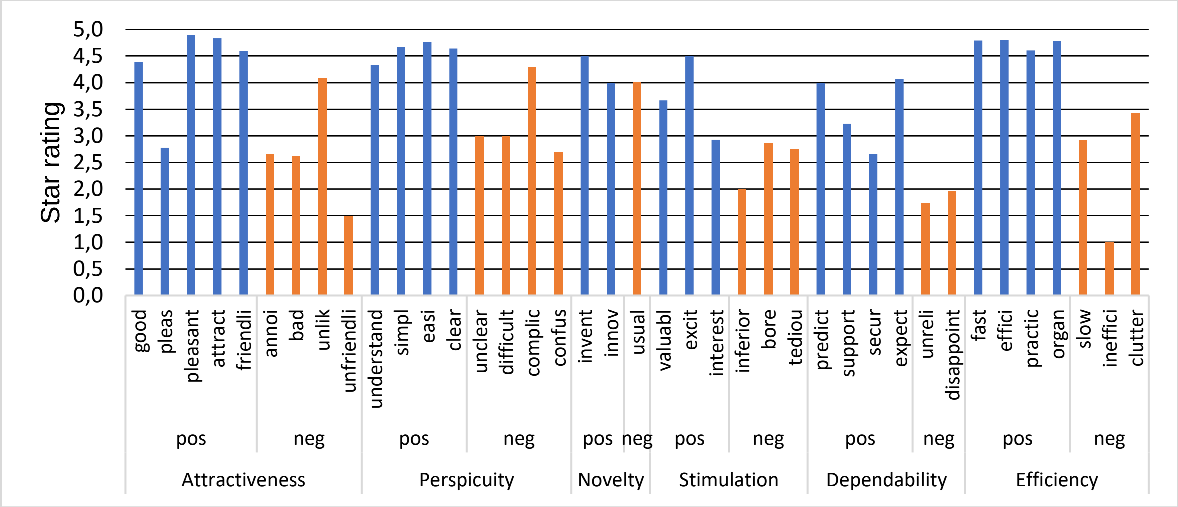

Figure 4. Linear weights of found UEQ words sorted by UEQ dimensions and subdimensions.

Table 6

Linear regression weights of found UEQ words in the dataset

UEQ dimension	UEQ subdimension	Found UEQ word	Linear regression weight
Attractiveness	positive	attract	4.83
		friendli	4.60
		good	4.39
		pleas	2.78
		pleasant	4.89
	negative	annoi	2.65
		bad	2.62

Table 6 (continued)

		unlik	4.09
		unfriendli	1.5
Perspicuity	positive	understand	4.33
		simpl	4.67
		easi	4.77
		clear	4.64
	negative	unclear	3
		difficult	3
		complic	4.29
		confus	2.69
Novelty	positive	invent	4.5
		innov	4
	negative	usual	4.02
Stimulation	positive	valuabl	3.67
		excit	4.5
		interest	2.93
	negative	inferior	2
		bore	2.86
		tediou	2.75
Dependability	positive	predict	4
		support	3.23
		secur	2.66
		expect	4.07
	negative	unreli	1.74
		disappoint	1.95
Efficiency	positive	fast	4.79
		effici	4.80
		practic	4.61
		organ	4.78
	negative	slow	2.92
		ineffici	1
		clutter	3.43

Since the semantic differentials were assessed separately under the premise that positive words were associated with higher star ratings and negative words with lower star ratings, it is important to test this assumption. Before testing for differences between positive and negative words for each UEQ dimension the normality assumption needed to be tested to decide whether to use parametric or non-parametric tests. Therefore, the Shapiro-Wilk test for normality was applied to each subdimension. The results can be seen in Table 7. A significant Shapiro-Wilk test is rejecting the null hypothesis that the linear weights are normally distributed, which was only the case for the positive subdimensions of Attractiveness and Efficiency. For the other subdimensions the test could either not be calculated because of small sample sizes or was not significant, which means that a normal distribution could be expected. For small group sizes (n < 20) it is recommended to use non-parametric tests (Bortz & Schuster, 2011) and since the group sizes were between 1 and 5 in this dataset, U-Tests were used in the following.

Table 7

Mean, standard deviation and Shapiro-Wilk test results for each UEQ subdimension

UEQ dimension	UEQ subdimension	*N*	*M*	*SD*	W	Normal distributed	*p*-value
Attractiveness	positive	5	4.30	.88	.75	no	.03*
	negative	4	2.71	1.06	.94	yes	.67
Perspicuity	positive	4	4.60	.19	.87	yes	.29
	negative	4	3.24	.71	.80	yes	.10
Novelty	positive	2	4.25	.35	-	-	-
	negative	1	4.02	-	-	-	-
Stimulation	positive	3	3.70	.79	.99	yes	.93

Table 7 (contined)

	negative	3	2.54	.47	.84	yes	.22
Dependability	positive	4	3.49	.67	.89	yes	.38
	negative	2	1.85	.15	-	-	-
Efficiency	positive	4	4.74	.09	.70	no	.01*
	negative	3	2.45	1.28	.90	yes	.38

Note. Missing values could not be calculated because of too small sample sizes

* *p* < .05.

U-test for each UEQ dimension

The U-test was administered six times, once for each dimension and tested whether the mean rank of positive UEQ words was higher than the mean rank of negative UEQ words. For each test the tested data were the regression weights of positive UEQ words (group 1) and of negative UEQ words (group 2). An alpha level of .05 was used. The *p*- and *U*-values can be seen in Table 8. The differences for the dimensions Attractiveness (*U*(6,4) = 1, *p* = .02), Efficiency (*U*(4,3) = 0, *p* = .03) and Perspicuity (*U*(4,4) = 0, *p* = .01) were significant. The tests for the dimensions Dependability (*U*(4,2) = 0, *p* = .07), Stimulation (*U*(3.3) = 0, p = .05) and Novelty (*U*(2,1) = 2, *p* = 1) were not significant. In other words, positive words were significantly higher in the first three dimensions than negative words. This is a finding that supports the assumptions of the current method. In contrast, the latter three dimensions did not show any difference between positive and negative words.

Table 8

U-Test results for each UEQ dimension

UEQ dimension	*U*-Statistic	*p*-value
Attractiveness	1	.02*
Efficiency	0	.03*
Perspicuity	0	.01*
Dependability	0	.07
Stimulation	0	.05
Novelty	NA	NA

Note. The groupsize in the Novelty dimension was too small to apply the *U*-Test (group 1 = 2 and group 2 = 1).
* *p* < .05.

After averaging the subdimension means in Table 7, six ratings were obtained, one for each UEQ dimension. The result can be seen in Figure 5. It shows that especially Dependability and Stimulation dimensions show lower ratings as the other dimensions.

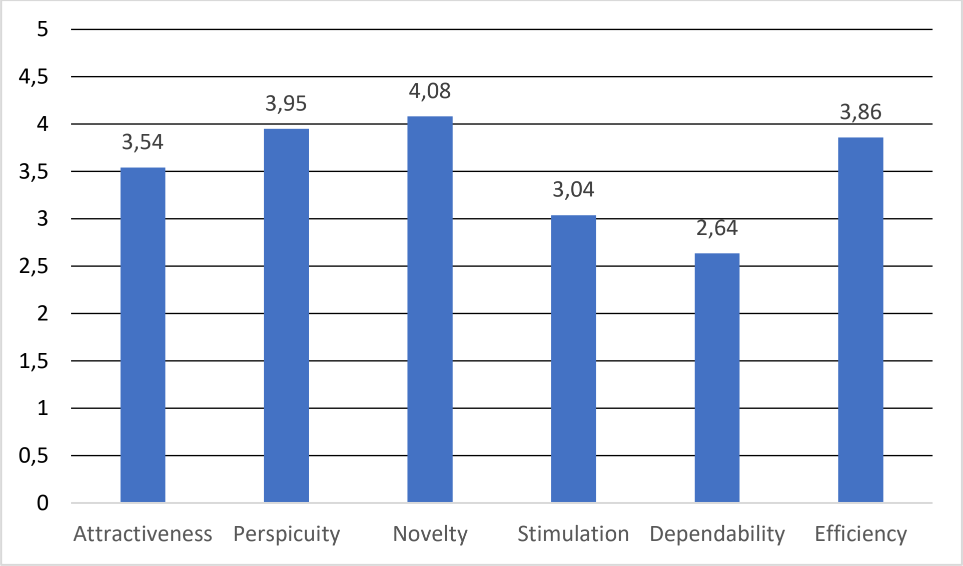

Figure 5. UEQ evaluation of the easyjet app based on NLP methods.

The goal of RQ1 was to assess the UEQ with NLP. Taking the NLP-evaluation results of the UEQ and transforming them to a scale with the range [-3;3] enabled the use of the UEQ data analysis tool (*User Experience Questionnaire (UEQ)*, 2019). Using this tool, a visual rating like the survey UEQ was created and is shown in Figure 6. Transforming the ratings in Figure 5 from a [1;5] scale to a [-3;3] scale results in a rating akin to a typical UEQ survey evaluation. Figure 6 presents the results of the current study in the same format as survey UEQ results are presented. Table 9 shows the UEQ values evaluated with the NLP method (NLP evaluation) and the linear transformation into a [-3;3] scale (Transformed evaluation).

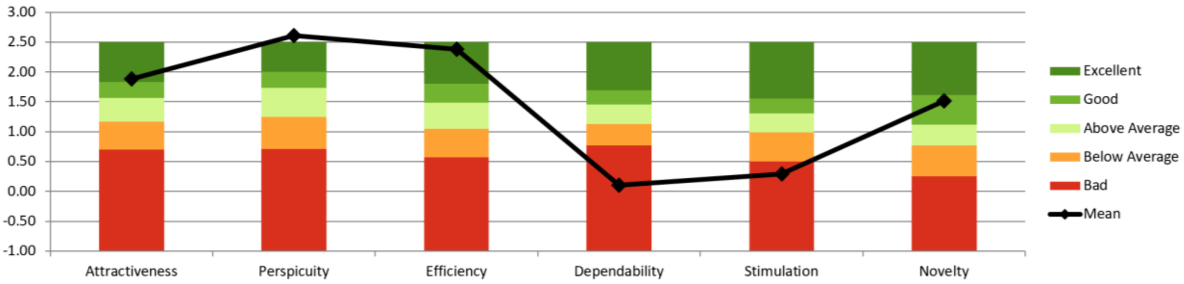

Figure 6. Transformed evaluation of the easyJet app shown in the UEQ benchmark.

Table 9

UEQ results assessed with NLP methods (1 to 5 scale) and the transformed values therefore (-3 to 3 scale)

UEQ dimension	NLP-evaluation	Transformed evaluation
Attractiveness	4.17	1.89
Efficiency	4.56	2.38
Perspicuity	4.72	2.61
Dependability	2.99	0.11
Stimulation	3.14	0.29
Novelty	4.04	1.52

Note. NLP-evaluation based on star ratings which have a scale range from 1 to 5. The Transformed evaluation is based on the NLP-evaluation but transformed to a scale ranging from -3 to 3.

Results for RQ 2

RQ2 examines the extent to which it is possible to rate app features with NLP, based on the UEQ. To be able to rate app features, first the co-occurrences of app features with UEQ words needed to be determined. The co-occurrences and the linear weights of UEQ words were then used to apply Equation (3) and rate each app feature. The results of this will be presented in the following.

Results of co-occurrences

App feature words refer here to the 27 stemmed words (column 3, Table 2) extracted from the app description representing features in the easyjet app. To assign UEQ words to app features, co-occurrences between the app feature words and UEQ words were determined. 370 co-occurrence pairs were found and are presented along with their frequency in Appendix C.

These 370 co-occurrence pairs are permutations of 25 unique app feature words (out of a total of 26 present app feature words) and 38 UEQ words (out of a total of 39 present UEQ words). Except for sport_equip, every app feature word was in the same UC as a UEQ word at least once. Some UEQ words were mentioned alone, as exemplified in the UC: "Annoying….".

Table 10 presents the co-occurrence pair frequency, once split by UEQ words and once by app feature words. This reveals that most co-occurrences were based on three app features (book, boarding_pass and check_in) as well as three UEQ words (easi, good and simpl). Interestingly, the top 3 app features are also the minimum interactions with easyJet app that a user needs to take a flight. Users need to book the flight, obtain a boarding pass and check in at the gate. This might suggest that the most occurred app features are the most important stages, from a user perspective, in the overall user journey. This can be seen in the UC: "fantastic app , one of the best out there , from booking to changing check in , everything works except my flightclub benefits" (5 star rating).

Further, the percentages of co-occurring app features based on their total occurrences in the dataset was calculated and is presented in column 3 of Table 10. These percentages can be seen as an indicator of how strongly users used UEQ words to describe these app features. For instance, 67.74% for "book" means that in 67.74% of the times book was mentioned a UEQ word was used in the same user comment. This is important information to consider when evaluating the co-occurrences. Based on the theory of semantic differentials (Osgood, 1952), very small percentages could for example hint to a low association of the app feature and the UEQ words. On average, in 57.67% of cases when an app feature was mentioned, it was accompanied by a UEQ word (*SD* = 11.42). This also indicates that users also use other words to talk about app features in 43.33% of the cases, as exemplified in the UC: “Cannot do much without data connection, not even display existing boarding pass. Should allow export to PDF. App does not render well on Nexus10”.

Table 10

Occurrences of UEQ words and app feature words in the analysis dataset.

App feature words	Frequency	Percentage of total occurrence	UEQ words	Frequency
book	3357	67.74	easi	2527
boarding_pass	2382	51.47	good	778
check_in	1599	54.41	simpl	504
card	379	53.42	pleas	191
gate	283	53.36	fast	182
view	277	46.21	secur	128
manag	214	75.7	clear	102
seat	207	58	annoi	90
passport	200	61.50	friendli	87
scan	188	70.21	bad	58
departur	111	44.14	confus	51
store	111	69.37	disappoint	46
bag	101	70.28	effici	43
change_flight	82	59.76	slow	38
arriv	73	43.84	difficult	36
offlin	63	55.56	expect	36
remind	62	88.71	support	34
filght_tracker	60	68.33	usual	33
camera	43	60.47	practic	28

Table 10 (continued)

live_updates	25	44	unlik	28
real_time	24	45.83	understand	25
locat	24	58.33	unreli	25
belt	15	46.67	interest	10
data_connection	9	44.44	excit	7
sport_equipment	3	-	complic	7
baggage_reclaim	2	50	bore	6
			pleasant	5
			clutter	4
			attract	4
			unfriendli	3
			organ	3
			inferior	3
			predict	2
			tediou	2
			ineffici	2
			innov	1
			unclear	1
			invent	1
Total	5131		Total	5131

Using the co-occurrence frequency and regression weights of every UEQ word for each app feature, the NLP-evaluation values were calculated using Equation (1) on an app feature level, which can be seen in Table 11. This evaluation is done on UEQ dimensions and informs researchers about how app features were perceived on different dimensions. Further, Table 11 shows that not all app features were rated on every dimension, i.e. Novelty and Stimulation have only few values. The reason for this are missing co-occurrences.

To derive the NLP-evaluation, the UEQ linear regression weights based on the whole UC dataset were used. It would have been possible to base the linear weights only on their co-occurrences with app features, but then the same UEQ word would have different weights for different app features. This way, the results between app features would not have been comparable.

The last row represents the app level NLP-evaluation that was obtained by averaging app feature level NLP-evaluation for each UEQ dimension. Figure 7 shows the resulting visual rating.

Table 11

NLP-evaluation on app feature levels.

App feature words	Attractiveness	Perspicuity	Novelty	Stimulation	Dependability	Efficiency
view	3.77	4.60	-	-	2.57	4.79
store	3.72	4.66	-	-	3.17	3.86
scan	4.14	4.65	4.02	-	2.66	4.58
real_time	3.81	4.47	-	-	-	4.79
passport	3.85	4.66	4.02	4.50	2.54	3.94
live_updates	4.39	4.77	-	-	-	-
gate	3.86	4.61	4.02	3.30	2.68	3.86
flight_tracker	3.88	4.74	-	2.93	2.89	4.28
data_connection	4.39	4.77	-	-	2.66	-
check_in	3.91	4.67	4.02	3.07	2.72	4.49
change_flight	3.96	4.70	-	-	2.61	4.70
camera	3.40	4.71	-	-	-	4.70
book	4.21	4.73	4.02	3.13	2.89	4.66
boarding_pass	3.88	4.68	4.07	3.15	2.69	4.23
belt	4.39	4.75	-	-	-	-
baggage_reclaim	-	4.77	-	-	-	-
card	4.03	4.63	4.02	-	2.63	4.14

Table 11 (continued)

seat	3.99	4.66	-	3.30	2.21	4.13
manag	4.15	4.70	4.02	-	3.34	4.62
bag	3.47	4.63	-	-	2.86	4.79
offlin	3.89	4.64	-	-	2.66	4.79
locat	3.69	4.77	-	-	3.23	-
arriv	2.65	4.73	-	-	1.85	2.92
departur	4.02	4.40	4.02	-	2.82	4.10
remind	4.38	4.58	-	-	3.36	4.28
Total NLP-evaluation	3.91	4.67	4.03	3.34	2.75	4.33

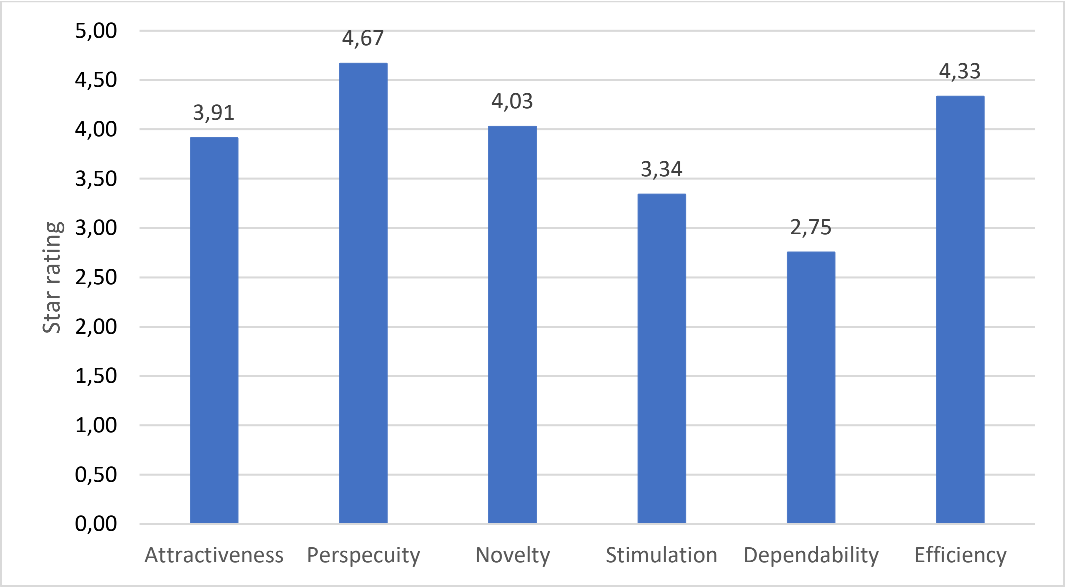

Figure 7. App level NLP-evaluation derived from app features ratings.

Discussion

So far, only few studies in UX research have used validated scales in combination with NLP methods (Aciar & Aciar, 2017; Lechler & Burghardt, 2017; Rodrigues et al., 2017). The present study used NLP methods to assess the UX of the easyJet Travel app in Google Play Store and combined it with a multiple linear regression. The findings suggest that an evaluation on the basis of the UEQ is possible. Additionally, it was possible to find specific app features and rate them on the same dimensions as the UEQ.

RQ 1

The assumption of the current method was that semantic differentials could be split into separate words and assessed separately with NLP methods. It was expected that the positive words would have higher ratings than the negative ones because semantic differentials measure polar terms and positive words of the UEQ are supposed to measure good UX. In turn, negative words should measure bad UX. The consolidated UEQ word weights for positive words were significantly higher than for negative words for the dimensions Attractiveness, Efficiency and Perspicuity and confirm the assumption. The result suggests that users use positive UEQ words more often when they describe experiences that they rate higher, as can be seen in the UC: “Well-designed, uncluttered, simple and easy to use. Great for booking planes quickly and checking flight details”, which was rated with 5 stars. No differences were observed for the dimensions Dependability, Stimulation and Novelty. An explanation for non-significance might lie in missing UEQ word weights because every dimension that did not show any difference had more than two missing values. The Novelty dimension had only one negative value which led to a p-value of .66. Despite non-significant tests, the present approach was able to extend the previous methods (Aciar & Aciar, 2017; Lechler & Burghardt, 2017; Rodrigues et al., 2017) and define a rating for each word not based on a fixed value or the authors subjective judgment, but empirically on a star rating assigned by users themselves. Just as these previous methods, the present approach also offers a great time saving in analyzing more than twenty thousand user comments.

Lastly, some UEQ words had unexpectedly high weights. This can be explained by negations (5 stars: “Does exactly what it says on the tin. Not complicated, even an idiot could use this without difficulties.”), word ambiguity (5 stars: “No problem loading and using this app. Unlike some!”) and stemming (1 star: “This app is so slow to load and I usually have to try 3 or 4 times to open it.”). This emphasizes the need to include rules addressing such issues in the method.

Although the NLP-evaluation showed that it is possible to rate an app on the UEQ dimensions, this does not mean that it is equivalent to the UEQ survey. Such a claim would require a direct comparison of applying the UEQ and the NLP method, which was out of the scope of the present thesis.

RQ 2

The present study was able to successfully rate app features on UEQ dimensions and enabled deeper insights into the rating of each app feature.

Most co-occurrences were in the Perspicuity and Attractiveness dimension. This raises the question of why these two dimensions were so heavily overrepresented. Possible explanations could be that these were particularly important dimensions that users paid attention to or valued extraordinarily. Alternatively, these words may just be used more commonly in everyday language. Linguistic research might offer answers in this regard.

The present approach was able to establish a UEQ evaluation for each co-occurring app feature, as can be seen in Table 11. In other words, each app feature had a unique combination of values of UEQ dimensions. This might prove beneficial for practice since it expands the information gained from the app and is also combined with the UEQ. Taking into account which app features were rated higher or lower, the present method provides insights into the initial question of why users rate an app the way they do.

Even though this seems promising, several challenges need to be considered.

First, this approach can also lead to counter-intuitive results, for instance, if both positive and negative words of a semantic differential co-occurred only once with an app feature and had the same star rating (e.g. 5 stars). If these were averaged, the result would be the highest, i.e. 5 star, rating for this app feature. Although this potential risk exists, it seems unlikely to occur with a sufficiently large dataset.

Second, 8 UEQ words were initially changed and were not the same as in the survey UEQ. This might have distorted the linear regression weights and co-occurrences, which might ultimately lead to different UEQ-evaluation for each app feature and a different overall rating of the app.

When comparing NLP-evaluations from RQ1 (Figure 5) and RQ2 (Figure 7), it becomes evident that the rating patterns (high Attractiveness/Perspicuity and low Dependability/Stimulation) resemble each other. This might seem obvious since the ratings of the app features in Figure 7 are based on UEQ weights and these also created the NLP-evaluation in RQ1. Yet, from this does not follow that a similar pattern will emerge since the constellation of UEQ and app feature co-occurrence can be quite unique as seen in Table 11 and include only certain UEQ words, which could lead to a very different NLP-evaluation on app level.

Although the present method was able to rate not only single app features, but also the whole app on each dimension and therefore create a similar rating as would have been done by the survey UEQ, it is unclear whether the results can be compared to the survey UEQ. In addition to the application of NLP methods, this would require conducting a usability test with subsequent UEQ surveys. This could prove a fruitful approach for future research.

This leads to the question of which NLP-evaluation (the ones based on all UEQ words, or on the app features) represents an equivalent to the survey UEQ. Possibly, the values based on all UEQ words capture the whole user journey and therefore the UX while the values based on the app features rather cover a type of usability of the app. An example of the wider scope would be the following quote from the UCs: "waste of time cost me my flight. followed live updates which said flight delayed when actually it had boarded and gate closed". The previous statement would contradict the idea of the UEQ which claims to capture both the user experience (hedonic dimensions) and usability (pragmatic dimensions), but user comments can have a wider scope than a survey since the survey is focused on the interaction with the app and often applied in the context of a usability test (Rauschenberger et al., 2013; Schrepp et al., 2014).

Limitations

Even though the results seem to be promising, several limitations of the present study need to be kept in mind. One limitation of the present method is missing values and the reason for them. Several types of missing values were found. First, some words were missing in the dataset because no user used them. Second, other words were present in the dataset, but were not found in the automated analysis because the algorithm could not correctly stem every word. For instance, the word "enjoyable" was not found, because it was stemmed to "enjoy" but words related to it (e.g. "enjoy") were stemmed to "enjoi" and „enjoi" has occurred 154 times.

Another drawback of stemming is that it might consolidate words that do not have the same meaning. An example is “good”, which included stemmed versions of words with totally different meaning like “goodbye” or “goods”.

In general, it is not clear if missing values are a sign for the absence of the particular dimension (e.g. Novelty) and therefore the total absence of the dimension in the app, or if users take this dimension as “given” or normal and therefore not noteworthy. Things that are taken for granted are presumably not noteworthy and are therefore not mentioned in the user comments. A different interpretation would be that the lack of data in a certain dimension is a sign of indecisiveness on the user side.

Further, even if words are found and co-occurrences are considered, the interpretation of these co-occurrences is not straightforward. For instance, the pair „easy – navig“ might offer hints about the navigation, but it remains unclear if the navigation inside the app is meant, or the navigation around the airport or the navigation of the aircraft (e.g. “Easy to use features and quick navigation to where you need to be”). To make this clear distinction, a more sophisticated approach would be needed or a manual check in the original user comment could be done to get more clarity about the meaning. This, in turn, would lower the scalability of the method. Although closed-vocabulary approaches make it relatively easy to count word occurrences, they ignore the context in which words are used and ambiguities to which they point (Kern et al., 2016).

Another limitation to the interpretation of the results is the timeframe of the data. The oldest user comment was posted 7 years ago. The app might have had different features at that time and therefore a different user experience. Analysing this data can be compared to analysing longitudinal data or taking a picture with long time exposure, as a result the picture gets blurry. Previous research has for example found that the emotionality of words in user comments of apps changes, even over a period of 12 months (Martens & Johann, 2017). Thus, it is very likely that user perceptions and therefore rating of the present app have changed.

A further limitation lies in the self-selection bias. Since making user comments is a voluntary act, this may result in only certain people choosing to comment and rating the app (Kern et al., 2016).

Apart from the problem of self-selection, the researcher’s subjectivity might influence the results as well. Although many steps of the current method are automated, selecting which app features to consider, choosing the words with which to describe the app features, deciding which words to consolidate or correct, and which to delete involves subjective decisions by the researcher that could influence the results.

The order of operations is another NLP-specific problem and important limitation. In the current study, stemming was done before deleting rare words. Both steps are important to consolidate words with the same meaning and include only relevant words. These steps can also be done in reverse order, but this will have a different impact on the results. If stemming is done before deleting rare words (n < 2), words with the same stem are merged, which increases their occurrence, but at the same time words can be merged that do not belong together from a semantic point of view (e.g. "booking" when booking a flight and "book" when reading a book). This risk could be reduced by doing it manually and bringing only related words together, but this would take a lot of time. Stemming after deleting rare words can lead to deletion of relevant words and loss of relevant data. Therefore, it is a compromise between accuracy, loss of relevant data, and time savings.

Lastly, one limitation to the findings of the present study are the occurrences of the word “easi”. Since the app and the company are called “easyJet”, this might have primed users and heightened the cognitive availability of the word “easy”, which in turn may have led to above-average use of this word (Farrell et al., 2012). An example for this could be: “easy to traverse, makes booking, check in and boarding easy too. easy app for easy jet flights.”

Contributions of the presented method

Despite the range of limitations, the present method has made several contributions. Through co-occurrences and regression weights it is possible to obtain additional information about the entire user journey. Another benefit is the large number of user comments. With the method developed in this thesis, it is possible to summarize thousands of user opinions in one instance. This means that the method is not only scalable, but also ecologically highly valid, since user opinions are not generated by the researcher asking for them or because an artificial laboratory situation requires it. Further, this method can theoretically be applied to any text. It would require adjusting the app feature words and the pre-processing, but the sequence of actions would not change.

Future work

Future research could focus on expanding the method by using topics instead of words as analysis element. Topics not only tend to be more informative than single words (Kern et al., 2016), but also more reliable (e.g. Diamantopoulos et al., 2012). A next step in the future would be to use cluster methods to test the hypothesized dimensional structure of the UEQ. Ideally, this would be done with a bigger dataset. I showed in this study that even a small dataset could lead to an information gain.

The most time-consuming steps in this study were the pre-processing steps because every token needed to be scanned, wrong tokens identified and replaced. This was a lot of manual work that could ideally be automated, potentially by machine learning.

Machine learning could not only be applied to support the pre-processing, but also to associate app features with UEQ words. The method could identify clusters and see if the UEQ words and app features are in the same cluster and if this is reflected in a standardized correlation or a similarity coefficient.

If the present study can be simplified and standardized, then it could bring an enormous value to the psychological field and connect it to other disciplines like NLP or ML.

Conclusion

The present study has extended the current literature by determining the UX of an app empirically, using a scalable method and based on previous UX research. The validated instrument UEQ was slightly modified to enable its use for natural language processing. Values for representing UEQ words were found and combined into means for each positive and negative subdimension of each UEQ dimension and finally to an evaluation value for each UEQ dimension. This is similar to the averaging of items in the survey UEQ and allows a global evaluation of the app on all 6 dimensions (Attractiveness, Perspicuity, Dependability, Efficiency, Stimulation and Novelty). Additionally, app features associated with the UEQ dimensions were identified and evaluated at a single semantic-differential level and at a global UEQ dimension level. In this way, the current approach allows the evaluation of an app feature on an UEQ-like scale. NLP has promising methods for working with natural language and using it for psychological research.

Appendix A, B and C

The appendices are too large for the print-out version. To save material, the appendix will be uploaded to a public gDrive and can be accessed via the following link: https://bit.ly/2QagPtU

References

Aciar, S., & Aciar, G. (2017). Analyzing User Experience Through Web Opinion Mining. In G. Meiselwitz (Ed.), *Lecture Notes in Computer Science: Vol. 10283, Social Computing and Social Media. Applications and Analytics* (pp. 203–214). Springer. https://doi.org/10.1007/978-3-319-58562-8_16

Atkins, D. C., Rubin, T. N., Steyvers, M., Doeden, M. A., Baucom, B. R., & Christensen, A. (2012). Topic models: A novel method for modeling couple and family text data. *Journal of Family Psychology*, *26*(5), 816–827. https://doi.org/10.1037/a0029607

Bargas-Avila, J. A., & Hornbæk, K. (2011). Old Wine in New Bottles or Novel Challenges? A Critical Analysis of Empirical Studies of User Experience. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 2689–2698). ACM. https://doi.org/10.1145/1978942.1979336

Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media Inc.

Bortz, J., & Schuster, C. (2011). *Statistik für Human-und Sozialwissenschaftler: Limitierte Sonderausgabe*. Springer-Verlag.

Calvo, R. A., Milne, D. N., Hussain, M. S., & Christensen, H. (2017). Natural language processing in mental health applications using non-clinical texts. *Natural Language Engineering*, *23*(05), 649–685. https://doi.org/10.1017/S1351324916000383

Cambria, E., & White, B. (2014). Jumping NLP Curves: A Review of Natural Language Processing Research. *IEEE Computational Intelligence Magazine*, *9*(2), 48–57. https://doi.org/10.1109/MCI.2014.2307227

Chen, E. E., & Wojcik, S. P. (2016). A practical guide to big data research in psychology. *Psychological Methods, 21*(4), 458–474. https://doi.org/10.1037/met0000111

Chen, N., Lin, J., Hoi, S. C. H., Xiao, X., & Zhang, B. (2014). AR-Miner: Mining Informative Reviews for Developers from Mobile App Marketplace. In P. Jalote, L. Briand, & A. van der Hoek (Eds.), *Proceedings of the 36th International Conference on Software Engineering* (pp. 767–778). ACM Press. https://doi.org/10.1145/2568225.2568263

Diamantopoulos, A., Sarstedt, M., Fuchs, C., Wilczynski, P., & Kaiser, S. (2012). Guidelines for choosing between multi-item and single-item scales for construct measurement: a predictive validity perspective. *Journal of the Academy of Marketing Science*, *40*(3), 434–449. https://doi.org/10.1007/s11747-011-0300-3

EasyJet: Travel App - Apps on Google Play. (2019, November 13). https://play.google.com/store/apps/details?id=com.mttnow.droid.easyjet

Farrell, M. T., Abrams, L., & White, K. (2012). The role of priming in lexical access and speech production. *Psychology of Priming*, 205–244.

Google Play Store: number of apps 2019. (2019, November 13). https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/

Graham, J., Haidt, J., Koleva, S., Motyl, M., Iyer, R., Wojcik, S. P., & Ditto, P. H. (2013). Chapter Two - Moral Foundations Theory: The Pragmatic Validity of Moral Pluralism. In *Advances in Experimental Social Psychology* (Vol. 47, pp. 55–130). https://doi.org/10.1016/B978-0-12-407236-7.00002-4

Ha, S. (2015). Assessing Quality of Consumer Reviews in Mobile Application Markets: A Principal Component Analysis Approach. In *PACIS 2015 Proceedings.* Symposium conducted at the meeting of PACIS.

Hassenzahl, M. (2004). The Interplay of Beauty, Goodness, and Usability in Interactive Products. *Human-Computer Interaction*, *19*(4), 319–349. https://doi.org/10.1207/s15327051hci1904_2

Hassenzahl, M., & Sandweg, N. (2004). From Mental Effort to Perceived Usability: Transforming Experiences into Summary Assessments. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems* (pp. 1283–1286). ACM.

Hassenzahl, M., Platz, A., Burmester, M., & Lehner, K. (2000). Hedonic and ergonomic quality aspects determine a software's appeal. In *the SIGCHI conference*, The Hague, The Netherlands.

History - The Human Factors and Ergonomics Society. (2019, November 13). https://www.hfes.org/about-hfes/hfes-history

Hoover, J., Dehghani, M., Johnson, K. M., Iliev, R. I., & Graham, J. (2019). Into the wild: Big Data Analytics in Moral Psychology. In K. Gray & J. Graham (Eds.), *Atlas of Moral Psychology*. Guilford.

Hornbaek, K. (2006). Current practice in measuring usability: Challenges to usability studies and research. *International Journal of Human-Computer Studies*, *64*(2), 79–102. https://doi.org/10.1016/j.ijhcs.2005.06.002

Infographic: The Biggest App Stores. (2019, November 13). https://www.statista.com/chart/12455/number-of-apps-available-in-leading-app-stores/

International Organization for Standardization (2018). *ISO 9241-11:2018.*

Kern, M. L., Park, G., Eichstaedt, J. C., Schwartz, H. A., Sap, M., Smith, L. K., & Ungar, L. H. (2016). Gaining insights from social media language: Methodologies and challenges. *Psychological Methods*, *21*(4), 507–525. https://doi.org/10.1037/met0000091

Langhe, B. de, Fernbach, P. M., & Lichtenstein, D. R. (2016). Navigating by the Stars: Investigating the Actual and Perceived Validity of Online User Ratings. *Journal of Consumer Research*, *42*(6), 817–833. https://doi.org/10.1093/jcr/ucv047

Laugwitz, B., Held, T., & Schrepp, M. (2008). Construction and Evaluation of a User Experience Questionnaire. In A. Holzinger (Ed.), *Proceedings of the 4th Symposium of the Workgroup Human-Computer Interaction and Usability Engineering of the Austrian Computer Society on HCI and Usability for Education and Work* (Vol. 5298, pp. 63–76). Springer-Verlag. https://doi.org/10.1007/978-3-540-89350-9_6

Law, E. L.-C., Roto, V., Hassenzahl, M., Vermeeren, A. P.O.S., & Kort, J. (2009). Understanding, scoping and defining user experience. In D. R. Olsen, R. B. Arthur, K. Hinckley, M. R. Morris, S. Hudson, & S. Greenberg (Eds.), *Chi 2009 - Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 719–728). ACM. https://doi.org/10.1145/1518701.1518813

Lechler, D., & Burghardt, M. (2017). User Experience Mining auf Basis von Online-Produktbewertungen. In M. Burghardt, R. Wimmer, C. Wolff, & C. Womser-Hacker (Chairs), *Mensch und Computer 2017-Tagungsband.* Symposium conducted at the meeting of Gesellschaft für Informatik e.V., Regensburg.

Lima, A. M. S., Silva, P. B. S., Cruz, L. A., & Mendes, M. S. (2017). Investigating the polarity of user postings in a Social System. In G. Meiselwitz (Ed.), *Lecture Notes in Computer Science: Vol. 10283, Social Computing and Social Media. Applications and Analytics* (pp. 246–257). Springer.

Maalej, W., Kurtanović, Z., Nabil, H., & Stanik, C. (2016). On the Automatic Classification of App Reviews. *Requirements Engineering*, *21*(3), 311–331. https://doi.org/10.1007/s00766-016-0251-9

Martens, D., & Johann, T. (2017). On the Emotion of Users in App Reviews: Old. In *2017 IEEE/ACM 2nd International Workshop on Emotion Awareness in Software Engineering (SEmotion)* (pp. 8–14). IEEE. https://doi.org/10.1109/SEmotion.2017.6

Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review*, *63*(2), 81–97. https://doi.org/10.1037/h0043158

Miller, G. A. (2003). The cognitive revolution: a historical perspective. *Trends in Cognitive Sciences*, *7*(3), 141–144. https://doi.org/10.1016/S1364-6613(03)00029-9

Osgood, C. E. (1952). The nature and measurement of meaning. *Psychological Bulletin*, *49*(3), 197–237. https://doi.org/10.1037/h0055737

Park, G., Schwartz, H. A., Eichstaedt, J. C., Kern, M. L., Kosinski, M., Stillwell, D. J., Ungar, L. H., & Seligman, M. E. P. (2015). Automatic personality assessment through social media language. *Journal of Personality and Social Psychology*, *108*(6), 934–952. https://doi.org/10.1037/pspp0000020

Persaud, I. (2010). Field Study. In *Encyclopedia of Research Design* (pp. 489–490). SAGE Publications, Inc. https://doi.org/10.4135/9781412961288.n152

Plank, B., & Hovy, D. Personality Traits on Twitter—or—How to Get 1,500 Personality Tests in a Week. In A. Balahur, E. van der Goot, P. Vossen, & A. Montoyo (Eds.), *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis* (pp. 92–98). Association for Computational Linguistics. https://doi.org/10.18653/v1/W15-2913

Rauschenberger, M., Schrepp, M., Perez-Cota, M., Olschner, S., & Thomaschewski, J. (2013). Efficient Measurement of the User Experience of Interactive Products. How to use the User Experience Questionnaire (UEQ). Example: Spanish

Language Version. *International Journal of Interactive Multimedia and Artificial Intelligence*, *2*(1), 39. https://doi.org/10.9781/ijimai.2013.215

Rodrigues, P., Silva, I. S., Barbosa, G. A. R., Coutinho, F. R. d. S., & Mourão, F. (2017). Beyond the Stars: Towards a Novel Sentiment Rating to Evaluate Applications in Web Stores of Mobile Apps. In R. Barrett (Ed.), *Proceedings of the 26th International Conference on World Wide Web Companion* (pp. 109–117). International World Wide Web Conferences Steering Committee. https://doi.org/10.1145/3041021.3054139

Sagi, E., & Dehghani, M. (2014). Measuring Moral Rhetoric in Text. *Social Science Computer Review*, *32*(2), 132–144.

Sajnani, A., Kamani, H., Jeswani, H., & Samdani, K. (2017). A Study on Natural Language Processing for Human Computer Interaction. *International Journal of Advanced Research in Computer Engineering & Technology*, *6*(11).

Schrepp, M., Hinderks, A., & Thomaschewski, J. (2014). Applying the User Experience Questionnaire (UEQ) in Different Evaluation Scenarios. In A. Marcus (Ed.), *Lecture Notes in Computer Science: Vol. 8517, Design, User Experience, and Usability* (pp. 383–392). Springer. https://doi.org/10.1007/978-3-319-07668-3_37

Schrepp, M., Hinderks, A., & Thomaschewski, J. (2017). Construction of a Benchmark for the User Experience Questionnaire (UEQ). *International Journal of Interactive Multimedia and Artificial Intelligence*, *4*(4), 40. https://doi.org/10.9781/ijimai.2017.445

Su, J., Shao, P., & Fang, J. (2008). Effect of incentives on web-based surveys. *Tsinghua Science and Technology*, *13*(3), 344–347. https://doi.org/10.1016/S1007-0214(08)70055-5

User Experience Questionnaire (UEQ): Data Analysis Tools. (2019, November 13). https://www.ueq-online.org/

Van Selm, M., & Jankowski, N. W. (2006). Conducting Online Surveys. *Quality and Quantity*, *40*(3), 435–456. https://doi.org/10.1007/s11135-005-8081-8

Yoganathan, D., & Sangaralingam, K. (2015). Designing Fitness Apps Using Persuasive Technology: A Text Mining Approach. In *PACIS 2015 Proceedings.* Symposium conducted at the meeting of PACIS.

Zhu, M., Zhao, F., Fang, X., & Moser, C. (2017). Developing Playability Heuristics Based on Nouns and Adjectives from Online Game Reviews. *International Journal of Human–Computer Interaction*, *33*(3), 241–253. https://doi.org/10.1080/10447318.2016.1240283

Universität
Basel

Fakultät für
Psychologie

FAKULTÄT FÜR PSYCHOLOGIE 2003 UNIVERSITÄT BASEL

Universität Basel, Fakultät für Psychologie, Missionsstrasse 60/62, 4055 Basel

Declaration of Authorship

I hereby declare - that I have written this Master's Thesis titled “The graphic designer was a 5 year old drawing with their toes”:
Applying Natural Language Processing to rate user comments of the easyJet Travel app by assessing the User Experience Questionnaire without any help from others and without the use of documents and aids other than those stated in the references, - that I have mentioned all the sources used and that I have cited them correctly according to the established academic citation rules.

First name, family name: Ewgeni, Wolkow

Matriculation number: 12-923-033

Date: 13.11.2019

Signature: ____________________________________