Information in Online Purchase Decision Processes:

Information Extraction, Data Analysis and Economic Impact

A dissertation submitted to attain the degree of

Dr. rer. pol.

by the School of Business, Economics and Information Systems of the University of Passau

submitted by

Tristan Wimmer

submitted on October 24, 2018

Tag der Disputation: 03.05.2019

Gutachter:
PD Dr. Michael Scholz
Prof. Dr. Franz Lehner

Acknowledgment

I want to thank several people without whom this thesis would not exist. First of all, I would like to thank my doctoral supervisor Prof. Dr. Michael Scholz. My PhD would not have been possible without his tremendous dedication and support. He assisted me with any question and did everything in his power to support me. His enthusiasm and knowledge was a major motivating factor for me. Our shared conversations about hobbies and interests provided me ample reward for my hard work.

Further, I want to thank Prof. Dr. Franz Lehner, who always promoted and supported me during my PhD. He always championed me throughout this opportunity to complete my PhD, as well as agreeing to be a member on my doctoral committee.

I also want to thank Claudia Reitmayer, who always helped me with organizational questions.

A special thank you goes to my former fellow student, Tobias Friedl, who assisted me to overcome my initial difficulties with informational science.

Additionally, I want to thank Prof. Dr. Jan Hendrik Schumann, who agreed to be a member of my doctoral committee.

Further I owe much gratitude to the Department of Information Systems as well as to the School of Business, Economics and Information Systems of the University of Passau for its support during my studies.

I also thank my former colleagues, namely, Alexandra Dzepina, Nora Fteimi, Sebastian Floerecke, Hans Achatz, Alexander Keller and Tobias Baumgärtner, for their mental support and Olga Ivanova for providing useful tips and tricks in the first months of my PhD.

A special thank you goes to my fiancée, Elisabeth Koll, who never lost confidence in me through the ups and downs. She has always supported me with her love and bolstered me to pursue my goals. I am happy to have her on my side for the rest of my life.

Most of all, I want to thank my family, - my sister Carina, my aunt Karin, as well as my deceased grandparents Elvira and Matthias. Especially, I want to thank my parents, Barbara and Gerro, who always assisted me and without whom I would not be where I am. Without my mum, supporting me at school, helping me with my homework for school in the afternoons for several years, I presumably would not have

finished my Abitur, nor would I be sitting here and writing this acknowledgment. I thank my dad, for the several discussions, in which he engaged critically with my plans and ideas and helped me bring them into reality. Without my family, this PhD would have never been possible. Thank you for everything you have done for me in the past, in the present, and in the future!

Contents

List of Figures

List of Tables

1 Introduction

Information has a particular importance in online purchase decision processes. As opposed to consumers in offline markets, consumers in online markets cannot inspect the physical product to evaluate it and reduce their perceived risk. Hence, consumers in online markets are dependent upon the information that they can gather about a product in which they are interested (Park and Lee, 2007). Therefore, they have two primary sources of information: Product descriptions and customer reviews. Both sources affect consumers' purchase decision processes and hence have an economic impact for consumers, shop providers and manufacturers. It is important to know how these sources of information influence a customer's online purchase decision and how to extract the relevant information for the following reasons: (1) to facilitate consumers in finding interesting and relevant information with low effort, (2) to utilize the dissolved information, (3) to support consumers during the purchase process and (4) to take economic advantage of the extracted information. These aims lead to several research objectives. This thesis answers the following research questions:

1. What classification method should be used depending on the data set characteristics?

2. How can product features be automatically extracted from online customer reviews?

3. How do product and review filtering systems influence purchase decision processes in online markets?

4. How do the presence and infomration amount of online product descriptions influence sales?

To examine these research objectives, several studies applying different methodological approaches have been conducted. Overall, this thesis consists of 4 studies, depicted in Figure 1.1

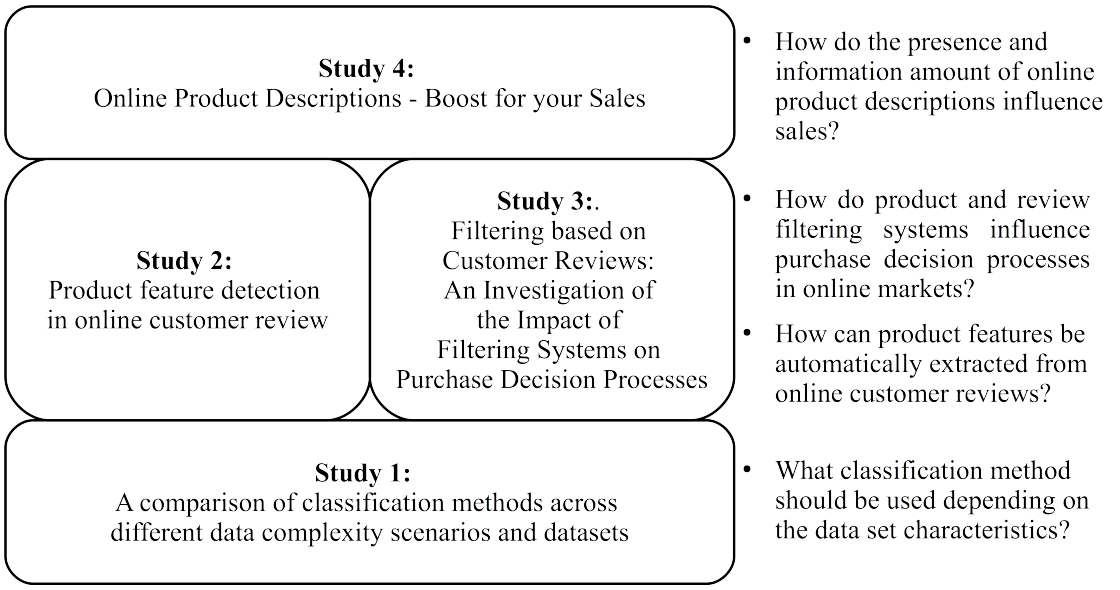

Figure 1.1: Overview of Research Studies

The first study compares 25 classification methods in four different complexity scenarios and on datasets described by five data characteristics. Thus, study 1 provides an algorithmic foundation for study 2. In study 2, a new ensemble classification approach is proposed to detect product features in online customer reviews with a focus on sparse data, e.g., from small or new online shops. Whereas study 2 focuses on the algorithm to detect product features, study 3 investigates the impact of filtering systems based on extracted product features on a three-stage purchase decision process. In addition to customer reviews, consumers also use product descriptions to evaluate products, and hence product descriptions might influence a product's popularity. Therefore, study 4 investigates the influence of product descriptions on a product's sales for three different product categories. In the following, I provide a short overview of each study.

Study 1: A comparison of classification methods across different data complexity scenarios and datasets

Authors: Michael Scholz and Tristan Wimmer

This study investigates the performance of 25 classifiers in four different complexity scenarios and on datasets described by five data characteristics. Classification problems arise in many areas of research (e.g., disease detection (Maysanjaya et al., 2015; Bourouhou et al., 2016), financial forecasting (Feldman and Gross, 2005; Lessmann et al., 2015; Fitzpatrick and Mues, 2016), and traffic sign detection (Kiran et al., 2009; Stallkamp et al., 2012)). Therefore, a large set of algorithms has been proposed in the existing literature, some of which have a focus on special classifica-

tion problems. There is evidence that data characteristics influence the accuracy of classification methods (Kiang, 2003; Shreve et al., 2011). In this study, we used a controlled setting in which we generated synthetic datasets to investigate the influence of six data characteristics on the performance of classification methods. We applied individual, homogeneous and heterogeneous classifiers in four complexity scenarios. This study shows that the performance of classification methods is significantly influenced by data characteristics. We found that heterogeneous ensemble classifiers reach the highest overall accuracy and examined the best approaches for each complexity scenario and dataset characteristic. Thus, this study assists researchers and practitioners in selecting the best classifier subject to the characteristics of their datasets.

Study 2: Product feature detection in online customer reviews
Author: Tristan Wimmer

Customers discuss their experiences about a product in online customer reviews. These reviews are a major source of information for other consumers in the purchase decision process to reduce consumers' uncertainty about whether a product fits their needs (Kang et al., 2017; Zhu and Zhang, 2010). Consumers tend to process all available information they can obtain about a product of interest (Park and Lee, 2007); however, the number of online customer reviews on e-commerce websites has increased substantially in recent years (Chen and Xu, 2017; Guo and Zhou, 2017). Nevertheless, consumers have a limited processing capacity and become cognitively overloaded if there are too many customer reviews to process, resulting, inter alia, in cognitive strain – the so-called information overload phenomenon (Jacoby et al., 1974; Keller and Staelin, 1987; Malhotra, 1984). Retailers can support customers in their purchase decision process with product filtering systems based on product features in customer reviews. Thus, consumers can filter product information and identify the sections that discuss the features in which they are interested. To enable retailers in offering such decision support systems, they must be able to automatically detect product features from customer reviews. This study proposes a new ensemble approach to detect product features from online customer reviews. In this paper, I investigate the performance of a bag-of-words model and a maximum entropy model for feature detection in online customer reviews. The results show that a bag-of-words model has high recall but also low precision, whereas the maximum entropy model has low recall and high precision. Thus, I combine both approaches into a new ensemble approach and show with three different datasets and product categories that the ensemble approach is superior to the other classification methods. Further, I demonstrate that the superior accuracy is constant even for small

training dataset sizes. The contribution of this study is to provide an algorithm to detect product features from customer reviews for even a few customer reviews per product category and thus is especially recommendable for new or small shop providers, providing them with the opportunity to offer services such as decision support systems based on product features in customer reviews.

Study 3: Filtering based on Customer Reviews: An Investigation of the impact of Filtering Systems on Purchase Decision Process
Authors: Tristan Wimmer and Michael Scholz

In this study, we investigate the impact of filtering systems on the purchase decision process. We followed Wu and Rangaswamy (Wu and Rangaswamy, 2003) and assumed a three-stage-purchase decision process. For our investigation, we conducted a laboratory experiment and measured the size, the objective quality as well as the average price of each stage of the purchase decision process (e.g., consideration set, choice set and choice). We investigated two types of filtering system. First, we used product filtering based on customer reviews. Depending on their treatment, the participants were able to filter products by selecting one or more product features. Thereby, only those products were listed whose customer reviews discuss the selected feature. As a second type of filtering, we implemented customer review filtering. The participants were able, depending on their treatment, to filter customer reviews, analogous to products, by selecting a product feature on a product detail page. Afterwards, only customer reviews discussing the selected feature are shown. We use a 2x2 full factorial design to test all combinations of product filtering, customer review filtering and no filtering. The results of our laboratory experiment show that both systems reduce consumers' effort in making a purchase decision and that product filtering reduces the amount of time consumed in the purchase decision making process. Furthermore, we examine whether a product filtering system ensures that the products considered in the early stages of the purchase decision process are already of high quality. This allows consumers to be more focused on a product's price than on its quality while evaluating a product, which ultimately leads to a selection of products that are significantly cheaper. The findings of this paper contribute to determining how retailers can benefit from offering filtering systems and provide a better understanding of how filtering systems influence each stage of the purchase decision process.

Study 4: Online Product Descriptions – Boost for your Sales?

Authors: Tristan Wimmer and Michael Scholz

Consumers in online as well as offline markets typically perceive uncertainty in purchase decision processes (Akerlof, 1970; Dimoka et al., 2012; Overby and Jap, 2009). Consumers in online stores predominantly have two sources of information to learn about a product's characteristics: customer reviews and product descriptions (Ghose, 2009). The existing research has intensively investigated the impact of customer reviews on sales; however, thus far, it has only sparsely analyzed the effect of product descriptions on sales. With this study, we address this gap in the research and examine the influence of product descriptions on sales. More precisely, we examine the effect of the presence as well as the amount of product descriptions on sales. Further, we distinguish between the effect due to product descriptions written by the retailer and descriptions written by the manufacturer. Further, this study investigates the interaction effect of product descriptions and customer reviews on sales. Based on empirical data from Amazon.com, we found that the existence of product description increases sales. Descriptions by Amazon.com have a slightly higher impact on sales than product descriptions written by the retailer. Further, the results show that product descriptions with more information have a higher impact on sales. Amazon-generated product descriptions contain more information and thus especially affect product sales with no or only a few customer reviews. Manufacturer-generated product descriptions, in contrast, have a higher impact on sales for products with many reviews. The study demonstrates that manufacturers and retailer should not only focus on customer reviews, but they should also provide product descriptions.

References

G Akerlof. The Market for " Lemons ": Quality Uncertainty and the Market Mechanism. *The Quarterly Journal of Economics*, 84(3):488–500, 1970.

A Bourouhou, A Jilbab, C Nacir, and A Hammouch. Comparison of classification methods to detect the Parkinson disease. In *2016 International Conference on Electrical and Information Technologies (ICEIT)*, pages 421–424, 2016.

Runyu Chen and Wei Xu. The determinants of online customer ratings: a combined domain ontology and topic text analytics approach. *Electronic Commerce Research*, 17(1):31–50, 2017. ISSN 15729362. doi: 10.1007/s10660-016-9243-6.

Angelika Dimoka, Yili Hong, and Paul A Pavlou. On Product Uncertainty in Online Markets: Theory and Evidence. *MIS Quarterly*, 36(2):395–426, 2012.

David Feldman and Shulamith Gross. Mortgage Default: Classification Trees Analysis. *The Journal of Real Estate Finance and Economics*, 30(4):369–396, 2005.

Trevor Fitzpatrick and Christophe Mues. An empirical comparison of classification algorithms for mortgage default prediction : evidence from a distressed mortgage market. *European Journal of Operational Research*, 249(2):427–439, 2016. ISSN 0377-2217.

Anindya Ghose. Internet EXCHANGES FOR U SED GOODS : AN EMPIRICAL ANALYSIS OF TRADE PATTERNS AND ADVERSE SELECTION. *MIS Quarterly*, 33(2):263–291, 2009.

Bin Guo and Shasha Zhou. What makes population perception of review helpfulness: an information processing perspective. *Electronic Commerce Research*, 17(4):585–608, 2017. ISSN 15729362. doi: 10.1007/s10660-016-9234-7.

Jacob Jacoby, Donald E. Speller, and Carol A. Kohn. Brand Choice Behavior as a Function of Information Load. *Journal of Marketing Research*, 11(1):63–69, feb 1974. ISSN 00222437. doi: 10.2307/3150994.

Mangi Kang, Jaelim Ahn, and Kichun Lee. Opinion mining using ensemble text hidden Markov models for text classification. *Expert Systems with Applications*, 0:1–10, 2017. ISSN 09574174. doi: 10.1016/j.eswa.2017.07.019.

Kevin Lane Keller and Richard Staelin. Effects of Quality and Quantity of Information on Decision Effectiveness. *Journal of Consumer Research*, 14:200–213, 1987. doi: 10.2307/2489411.

Melody Y Kiang. A comparative assessment of classification methods. *Decision Support Systems*, 35(4):441–454, 2003.

C G Kiran, L V Prabhu, V Abdu Rahiman, and K Rajeev. Traffic sign detection and pattern recognition using support vector machines. In *Seventh International Conference on Advances in Pattern Recognition*, pages 87–90, 2009.

Stefan Lessmann, Bart Baesens, Hsin-Vonn Seow, and Lyn C Thomas. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, 247(1):124–136, 2015.

Naresh K. Malhotra. Reflections on the Information Overload Paradigm in Consumer Decision Making. *Journal of Consumer Research*, 10:436–440, 1984. doi: 10.2307/2488913.

I M D Maysanjaya, H A Nugroho, and N A Setiawan. A comparison of classification methods on diagnosis of thyroid diseases. In *2015 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, pages 89–92, 2015.

E. Overby and S. Jap. Electronic and Physical Market Channels: A Multiyear Investigation in a Market for Products of Uncertain Quality. *Management Science*, 55(6):940–957, 2009. ISSN 0025-1909.

Do-Hyung Park and Jumin Lee. eWOM overload and its effect on consumer behavioral intention depending on consumer involvement. *Electronic Commerce Research and Applications*, pages 386–398, 2007. doi: 10.1016/j.elerap.2007.11.004.

J Shreve, H Schneider, and O Soysal. A methodology for comparing classification methods through the assessment of model stability and validity in variable selection. *Decision Support Systems*, 52(1):247–257, 2011.

J Stallkamp, M Schlipsing, J Salmen, and C Igel. Man vs. computer: {Benchmarking} machine learning algorithms for traffic sign recognition. *Neural Networks*, 32: 323–332, 2012.

Jianan Wu and Arvind Rangaswamy. A Fuzzy Set Model of Search and Consideration with an Application to an Online Market. *Marketing Science*, 22 (3):411–434, 2003. ISSN 07322399. doi: 10.1287/mksc.22.3.411.17738. URL `http://mktsci.journal.informs.org/content/22/3/411.abstract`.

Feng Zhu and Xiaoquan (Michael) Zhang. Impact of Online Consumer Reviews on Sales : The Moderating Role of Product and Consumer. *Journal of Marketing*, 74 (2):133–148, 2010. ISSN 0022-2429. doi: 10.1509/jmkg.74.2.133.

2 A comparison of classification methods across different data complexity scenarios and datasets

Abstract

Recent research assessed the performance of classification methods mainly on concrete datasets whose statistical characteristics are unknown or unreported. The performance furthermore is often determined by only one performance measure, such as the area under the receiver operating characteristic curve. We compare the performance of several classification methods in four different complexity scenarios and on datasets described by five data characteristics. We synthetically generate the datasets in order to control the statistical characteristics of each dataset. The performance of each classification method is determined by six measures. Our investigation reveals that heterogeneous classifiers perform best on average and that some classifiers are especially recommendable in a particular scenario or for training data with particular properties. We furthermore present insights into the relations among several performance measures.

Authors: Michael Scholz, Tristan Wimmer

2.1 Introduction

Classification is the problem of assigning observations to a predefined set of categories (i.e., classes). The observations are described by several data that are called features or explanatory variables. As classification methods require a training set of data containing observations for whom the correct class is known, these methods are considered as supervised learning methods (Hastie et al., 2008). A classification method (often called classifier) aims at identifying patterns that assign the observations in the training set to their corresponding classes. The patterns can then be applied to novel data in order to predict class assignments. Existing methods differ in the way the patterns are expressed and identified. For example, a linear discriminant analysis expresses patterns as linear combinations of features whereas support vector machines strive to identify a margin between two classes of data that is as large as possible. Because classification problems arise in many areas of

research (e.g., disease detection (Maysanjaya et al., 2015; Bourouhou et al., 2016), financial forecasting (Feldman and Gross, 2005; Lessmann et al., 2015; Fitzpatrick and Mues, 2016), and traffic sign detection (Kiran et al., 2009; Stallkamp et al., 2012)), a variety of methods has been proposed in the last decades that implement different ideas of how to express and assign patterns that allow assigning classes to observations. Researchers and practitioners are thus faced with the problem of selecting an adequate classification method for their classification problem.

There is ample evidence that the data generation process and hence the data to be classified significantly influence the accuracy with which classes will be assigned to observations (Kiang, 2003; Shreve et al., 2011). Some classification methods explicitly assume a certain data generation process. Logistic regression, for example, assumes that features are unimodally distributed and linear discriminant analysis assumes normally distributed features (Kiang, 2003). The selection of an adequate classification method also depends on the amount of available observations in the training set. Methods such as distance weighted discrimination are tailored towards high dimensional but low sample size problems (Marron et al., 2007). Existing comparative studies investigate the performance of classification methods mostly on one or a few datasets whose data generation process is rather unknown (e.g., Chen, 2011; Lessmann et al., 2015; Fitzpatrick and Mues, 2016). Very few studies utilize synthetic datasets to control for several characteristics of the dataset (Kiang, 2003; Asjad et al., 2018)). None of the existing studies, however, controls for both the data generation process and the data complexity (i.e., sample size and dimensionality) simultaneously.

In this study, we use a controlled setting in which we generate synthetic datasets to investigate the effect of several characteristics of the dataset on the performance of classification methods in four major complexity scenarios: low dimensionality and low sample size (LDLSS), low dimensionality and high sample size (LDHSS), high dimensionality and low sample size (HDLSS), and high dimensionality and high sample size (HDHSS). We compare individual classification methods and ensemble classification methods and use an adaptive grid search to find optimal classifier parameters (e.g., the number of trees for a random forest). We rely on the case of a binary classification in this study. Our study contributes to better understand the characteristics of classification methods and guide researchers and practitioners in selecting classification methods being appropriate for their binary classification problem.

The remainder of this paper is organized as follows: We review relevant comparative studies in the next section. We briefly introduce the classification methods used in this study in Section 2.3. The experimental design of our comparative assessment is

described in Section 2.4 and the results are presented in Section 2.5. We conclude this paper with a discussion of the results and implications for practitioners and researchers.

2.2 Related Work

Two streams of comparative studies are relevant for this study. First, studies comparing classification methods on real data and second, studies comparing classification methods on synthetic data. While the first stream of research contributes to identifying classification methods that are appropriate for a particular application (e.g., credit scoring), the second stream of research helps to understand the data characteristics that drive the performance of the investigated classification methods.

A plenty of studies exist comparing two or more binary classification methods on real-world data. Some of these studies intend to compare a proposed classification methods to other methods on several datasets in order to study the proposed classifier's performance (e.g., Eibe et al., 1998; Breiman, 2001; Sun et al., 2016). These studies compare a rather small number of classification methods and oftentimes classification methods using similar algorithms. Eibe et al. (1998), for example, compared the performance of decision tree algorithms to the performance of a linear regression and demonstrate that C5.0 and M5' significantly more often outperform a logistic regression than vice versa. Tibshirani et al. (2003) developed the nearest shrunken neighbor classifier and compare it especially to linear discriminant analysis. Their results demonstrate that the nearest shrunken neighbor classifier is superior to the linear discriminant analysis in terms of the classification error. Sun et al. (2016) proposeed a stabilized nearest neighbor classifier and compare their proposed method to three other classifiers of type nearest neighbor estimator. They show on different datasets that the stabilized nearest neighbor classifier produces more stable results than other nearest neighbor methods.

Other studies using real-world data aim at comparing a rather large set of classification methods for a specific application. For example, various studies compare classification methods for credit scoring (e.g., Baesens et al., 2003; Finlay, 2011; Marqués et al., 2012; Kruppa et al., 2013; Lessmann et al., 2015; Fitzpatrick and Mues, 2016; Maldonado et al., 2017). Recent results indicate that especially ensembles that are based on different classification methods perform best in credit scoring (Finlay, 2011; Lessmann et al., 2015). Ensembles joining the results of homogeneous classifiers (i.e., ensemble classifiers that aggregate multiple results from the same classification method) perform better than individual classifiers on average. Especially random forests were identified as rather accurate (Marqués et al., 2012; Kruppa et al., 2013; Lessmann et al., 2015). Interestingly, logistic regression has

been found to be one of the best individual classifiers in several studies (West, 2000; Marqués et al., 2012; Lessmann et al., 2015). Maldonado et al. (2017) propose an adaptation of a linear support vector machine and show that their method outperforms a logistic regression and the Fisher score and other methods based on support vector machines.

A similar application area for which classification methods have been investigated is the forecasting of bankruptcy (de Andrés et al., 2005; Li et al., 2011; Olson et al., 2012; Wang et al., 2014a; du Jardin, 2016, 2018). Studies in this area also indicate that ensemble methods perform better than individual classifiers (Kim and Upneja, 2014; Wang et al., 2014a; du Jardin, 2016). In a comparative study of several individual classifiers, Lahmiri (2016) showed that support vector machines with polynomial kernel function have the highest performance. Van Gestel et al. (2006) achieved the highest performance with a Bayesian kernel-based support vector machine compared to a linear discriminant analysis and logistic regression.

Binary classification methods also have been widely applied to sentiment classification (Xia et al., 2011; da Silva et al., 2014; Wang et al., 2014b; Wan and Gao, 2015). In a comparative study of some individual and ensemble classifiers, Wang et al. (2014b) show evidence that support vector machines perform best in sentiment classification. This is in line with the results of Pang et al. (2002); Tan and Zhang (2008) and partially with the results presented by Su et al. (2012). In contrast, Whitehead and Yaeger (2008); Xia et al. (2011); da Silva et al. (2014) and Wan and Gao (2015) obtained the highest performance with ensemble classifiers rather than support vector machines. Naïve Bayes classifiers have been furthermore shown to predict the sentiment of texts rather accurately (Xia et al., 2011; Wang et al., 2014b; Wan and Gao, 2015) whereas the k-nearest neighbor classifier has been found to perform worst (Tan and Zhang, 2008; Wang et al., 2014b).

Comparative studies have been also published for several other application areas, such as student retention prediction (Delen, 2010), census income classification (Caruana and Niculescu-Mizil, 2006), Internet traffic flow classification (Park et al., 2006; Williams et al., 2006), phishing detection (Abu-Nimeh et al., 2007), and disease and lesion detection (Dreiseitl et al., 2001; Baumgartner et al., 2004; Sajda, 2006; Aruna et al., 2011; Odeh and Baareh, 2016). Support vector machines have been identified as a highly and often most accurate classifier in several of these studies (e.g., Dreiseitl et al., 2001; Baumgartner et al., 2004; Sajda, 2006; Delen, 2010; Aruna et al., 2011; Pineda et al., 2015). Ensemble classifiers, if included in the studies, also showed a high and mostly the best performance (Park et al., 2006; Das and Sengur, 2010; Mohebian et al., 2017). Logistic regression has been also found to be amongst the top performing classifiers in some studies (Dreiseitl et al., 2001; Pineda

et al., 2015).

Comparative studies using real-world data hence indicate that the most promising classifiers are ensemble classifiers and support vector machines with respect to their predictive performance. These studies have been conducted on different dataset with a different complexity and typically unknown data characteristics (e.g., feature distributions) and different performance measures. It is hence not possible to generalize the findings from these studies.

A second stream of literature compares classification methods on synthetic dataset. The major advantage of this approach is the possibility to control the complexity and characteristics of the generated datasets. This, however, comes with the disadvantage that real-world datasets might have characteristics that are different from the controlled characteristics of synthetic datasets. According to this disadvantage, only a few comparative studies have been carried out so far on synthetic datasets.

We found two types of studies comparing classifiers on synthetic data: studies proposing a method that is compared to a few other methods (e.g., Ng, 2004; Marron et al., 2007; Sun et al., 2016) and studies comparing multiple classifiers without focusing on a particular method. For example, Marron et al. (2007) compared their proposed distance weighted discrimination method to support vector machines and regularized logistic regression on simulated data. They systematically varied the dimensionality of simulated data in order to demonstrate in which scenarios a distance weighted discriminator outperforms a support vector machine and a regularized logistic regression. Ng (2004) compared two types of a regularized logistic regression on synthetic data and systematically varied the number of features and the number of observations used for training. He found that an L_1 regularization is superior to an L_2 regularization especially in a high-dimensionality low-sample size scenario. Sun et al. (2016) compared their proposed stabilized nearest neighbor classifier to other nearest neighbor classifiers on synthetic as well as real-world data. The synthetic data were generated in order to test different distributions of the two classes. With their evaluations, Sun et al. (2016) demonstrated that their proposed classifier provides classification solutions that are more stable (i.e., reproducible) than those of other nearest neighbor classifiers.

Very few studies exist that compare several classification methods on synthetic data without proposing a particular method. Kiang (2003) investigated the misclassification error of five classification methods on synthetic data with different characteristics, such as normally distributed features vs. non-normally distributed feature. She identified neural nets and logistic regression as those with the highest performance in most cases. Ensemble methods and support vector machines which have often been found to be among the top performing classifiers in studies based on real-world

data were not included in the study by Kiang (2003). Support vector machines and k-nearest neighbor classifier have been found to be the most accurate methods in a study by Entezari-Maleki et al. (2009). They compared seven classification methods with respect to the area under the receiver operating characteristic (ROC) curve and with a varying number of observations in the training set and a varying number of features. Ensemble methods were also not investigated in this study. Asjad et al. (2018) compared discriminant analysis, artificial neural networks and support vector machines on a synthetic dataset and showed that the support vector machine outperforms the other two classifiers. This study is very limited in the number of classification methods compared. Furthermore, the authors did not investigate the influence of different data characteristics, used only one performance measure and did not include ensemble methods in their analysis.

Although many comparative studies exist for the binary classification problem, most of these studies focused on a specific application area. The application area might determine the number of features and at least some data characteristics. Studies on synthetic datasets aim at identifying classification methods that show a high performance in general. Existing comparisons on synthetic datasets use only one performance measure, do not investigate ensemble methods or aim at proposing a particular classification method. Our study compares a large variety of classification methods – including ensemble methods – in four different complexity scenarios and six cases with different data characteristics. We furthermore compare the classifiers with six performance measure in order to overcome the disadvantages of a single performance measure.

2.3 Classification Methods

In this section, we discuss the classification methods investigated in this study. We distinguish between individual and ensemble classifiers. In total, we compare 25 classification methods. We selected established and often investigated methods (e.g., linear discriminant analysis, logistic regression, classification and regression tree) as well as novel methods (e.g., stabilized nearest neighbor, distance weighted discrimination). We limit the number of classifiers to 25 in order to keep our investigation tractable and computable within a few days[1]. Due to the high number of methods, it is not possible to describe the methods in detail. We hence only briefly describe the main idea of each classifier. Table 2.1 gives a summary of all classification methods analyzed in this study and shows the number of parameters that characterize each method. Each of the classifiers maps a feature vector X to class labels $y \in 1, \ldots, C$. As the focus of this study is on binary classification methods, we set $C = 2$.

[1] We discuss our experimental setting and the maximal average time allowed per classifier in Section 2.4.3

Table 2.1: Classification methods considered in our study

Type	Classification Method	Acronym	# Parameters
Individual	Linear Discriminant Analysis	LDA	1
	Regularized Discriminant Analysis	RDA	2
	Logistic Regression	LR	0
	Regularized Logistic Regression	RLR	3
	Bayesian Logistic Regression	bayLR	0
	k-Nearest Neighbor	kNN	1
	Nearest Shrunken Neighbor	NSN	1
	Stabilized Nearest Neighbor	SNN	1
	Naïve Bayes	NB	3
	CART (Classification And Regression Tree)	CART	1
	C5.0	C5.0	3
	Support Vector Machine linear	SVM_L	1
	Support Vector Machine polynomial	SVM_P	3
	Support Vector Machine radial	SVM_R	2
	Distance Weighted Discrimination linear	DWD_L	2
	Distance Weighted Discrimination polynomial	DWD_P	4
	Distance Weighted Discrimination radial	DWD_R	3
Ensemble	Bagged CART	bCART	0
	Random Forests	RF	1
	Boosted Logistic Regression	booLR	1
	Gradient Boosted Trees	GBT	4
	Simple Average All	SA_A	0
	Simple Average TOP3	SA_3	0
	Simple Average TOP5	SA_5	0
	Simple Average TOP7	SA_7	0

2.3.1 Individual Classifiers

Individual classifiers train one classification model to predict the assignment of observations to classes. Individual classifiers use either of two approaches to assign an observation i that is characterized by a vector X_i of feature levels to any of the two classes $y = 0$ and $y = 1$ in a binary classification problem. First, some methods (e.g., logistic regression) directly estimate the probability that X_i belongs to class $y = 1$. Other methods (e.g., support vector machines) estimate class conditional functions or probabilities to distinguish between observations for class $y = 0$ and observations for class $y = 1$.

2.3.1.1 Linear Discriminant Analysis

The **linear discriminant analysis** is a generalization of Fisher's linear discriminant algorithm (Fisher, 1936) and assumes that a hyperplane (discriminator or discriminant function) linear in the feature vector separates the two classes. This assumption only holds in the case of normally distributed features and a common covariance matrix (Hastie et al., 2008). The hyperplane maximizes the difference between the observations of the two classes and is described by a linear function.

Friedman (1989) proposed a **regularized discriminant analysis** that allows for non-common covariances and shrinking the separate covariances toward a common covariance.

2.3.1.2 Logistic Regression

Logistic regression is one of the most applied methods to the binary classification problem (Lessmann et al., 2015). It estimates a predictor function as linear combination of the feature variables and ensures that the linear combination is in $[0, 1]$. The predictor function also is a linear hyperplane that separates observations of class $y = 0$ from observations of class $y = 1$. A logistic regression is hence similar to a linear discriminant analysis, only the way the hyperplane is estimated is different between these two classification methods. A logistic regression requires that there are many more observations in the training set than features in order to robustly estimate the parameters of the predictor function (Ng, 2004). Regularization can also be used in logistic regressions in order to select and shrink the parameters of the predictor function. **Regularized logistic regressions** especially perform better than a logistic regression in high dimensional scenarios (Ravikumar et al., 2010). A logistic regression finds parameters that maximize the likelihood for the observations in the training set. Prior beliefs about the parameters are multiplied with the most likely parameter levels (maximum likelihood) in a **Bayesian logistic regression** (Gelman et al., 2008).

2.3.1.3 Nearest Neighbor Classifier

Nearest neighbor classifiers assume that observations belonging to the same class are densely distributed in the feature space. These classifiers use the nearest neighbors of an observation as prototypes to predict the class assignment for that observation (Hastie et al., 2008). **k-nearest neighbor classifiers** as the simplest approach of nearest neighbor classifiers predict class assignments based on the k most closely located training data in the feature space (Cover and Hart, 1967). Tibshirani et al. (2003) proposed a **nearest shrunken neighbor classifier** that shrinks each of the two class centroids toward the overall centroid of both classes by a certain amount (i.e., threshold). Sun et al. (2016) demonstrate that the classification results of k-nearest neighbor classifiers are likely to be unstable in a way that they cannot be reproduced with slightly perturbed data. They propose a **stabilized nearest neighbor classifier** that minimizes the classification instability that is the expected distribution of the distance between two classifiers.

2.3.1.4 Naïve Bayes Classifier

A **Naïve Bayes classifier** estimates a probability for each feature to occur in an observation that is assigned to class $y = 1$. By assuming that the features are

conditionally independent of each other, we get a simple likelihood function based on the conditional feature probabilities. Conditional probabilities of the two classes are then computed using the Bayes theorem as multiplication of the likelihood function and the prior class probabilities. Although features often are not independent of each other, Naïve Bayes classifiers are widely used due to their computational ease and their applicability also to scenarios with high dimensionality (Hastie et al., 2008).

2.3.1.5 Classification Trees

Classification trees estimate a sequence of linear splits of the feature space. The splits partition the feature space in rectangles that can be best described by one of the two classes. Splits are identified by measuring the homogeneity of and across the resulting partitions. The **classification and regression tree (CART)** algorithm was introduced by Breiman et al. (1984) and uses the Gini impurity to measure the homogeneity of a rectangle in the feature space with regard to the class variable y. Other classification tree algorithms (ID3, C4.5, C5.0) use the information gain as homogeneity measure. **C5.0** improves C4.5 in terms of memory usage and speed and C4.5 (Quinlan, 1993) is the successor of ID3 (Quinlan, 1986), so that we only investigate the performance of C5.0 in this study.

2.3.1.6 Support Vector Machines

Support vector machines (SVMs) are large-margin classifiers which aim at estimating a hyperplane in the feature space that separates the two classes and has a maximal distance to the nearest observations (Boser et al., 1992; Cortes and Vapnik, 1995). If the training data is linearly separable, we do have a **linear SVM** (SVM with linear kernel). A linear separation is often not very accurate. SVMs can estimate hyperplanes that are non-linear in the original feature space by applying the kernel trick proposed by Aizerman et al. (1964). In this study, we investigate two types of non-linear SVMs, **SVMs with radial basis function kernel** and **SVMs with polynomial kernel**. With the kernel trick SVMs estimate a linear hyperplane in a higher-dimensional feature space which comes with the disadvantage of an increased generalization error especially in scenarios with high dimensionality and low sample size (Marron et al., 2007; Jin and Wang, 2012).

2.3.1.7 Distance Weighted Discrimination

Distance weighted discrimination (DWD) is an approach that is similar to support vector machines but reduces the generalization error especially in high-dimensional low-sample size scenarios. Instead of maximizing the minimal distance to the hyperplane, DWDs minimize the average inverse distance (Marron et al., 2007; Qiao

et al., 2010). DWDs are an appealing approach in scenarios with high dimensionality and low sample size but come with the major disadvantage that they require solving a more complex optimization problem than SVMs do (Marron et al., 2007). DWDs are thus not well suited to be applied in scenarios with high sample size. In this study, we investigate the performance of **DWDs with linear kernel**, **DWDs with radial basis function kernel**, and **DWDs with polynomial kernel**.

2.3.2 Ensemble Classifiers

Ensemble classifiers estimate multiple classification models and either combine these models (e.g., boosted logistic regression) or their predictions (e.g., random forests). There is ample theoretical and empirical evidence that an ensemble of classification methods often increases the classification accuracy compared to an individual classifier (Dietterich, 2000). Ensemble methods therefore combine either models generated with the same algorithm (e.g., random forests) or models generated with different algorithms (e.g., simple average ensemble).

2.3.2.1 Bagging

Bagging (Bootstrap Aggregating) is an approach that mainly aims at decreasing the variance of the predictive performance of a classifier by using a bootstrapping technology to generate additional data for training. The same classifier is used to estimate a classification model for each bootstrapped training sample. The results of the classification models are finally averaged to assess the total performance of a bagged classifier. In this study, we assess the predictive performance of a **bagged CART** algorithm and **random forests**. In contrast to bagged CART, random forests select a subset of features at random in each bootstrapped training sample (Breiman, 2001).

2.3.2.2 Boosting

Boosting is an approach that aims at improving the predictive performance of a classification method by incrementally estimating classification models. Training data that a model misclassified are used to train a classification model in the next iteration. Like bagging, boosting also uses the same classifier for each classification model. One of the most often applied boosting strategies is gradient boosting which relies on the relation between the classification error and the gradient of a classification model (Cai et al., 2009). We include a **boosted logistic regression** (Friedman et al., 2000) and a **boosted classification tree** (Friedman, 2002) in our study.

2.3.2.3 Simple Average

Both, bagging and boosting, build ensembles with classification models that are generated by the same classification method. They are hence homogeneous ensemble

classifiers. Another strategy of building an ensemble is combining models and/or predictions of different classification methods. Different classification methods correctly classify data with different characteristics. Because a dataset can be seen as a mixture of data with different characteristics, it is obvious to assume that an ensemble of heterogeneous classifiers will perform better than an ensemble of homogeneous classifiers. A simple but highly accurate strategy of combining the results of multiple heterogeneous classifiers is to average their predictions (Lessmann et al., 2015). We include four heterogeneous simple average classifiers in our study: **simple average over all classifiers**, **simple average over the top 3 classifiers**, **simple average over the top 5 classifiers**, and **simple average over the top 7 classifiers**. The top performing classifiers are determined using the performance measures described in the following section.

2.4 Experimental Design

The focus of this paper is to compare classification methods across different complexity scenarios with respect to different characteristics of the data generation process. In this section, we describe the complexity scenarios, the data characteristics, the experimental setting, and the performance measures.

2.4.1 Complexity Scenarios

We approximate data complexity by two variables: sample size of the training dataset and number of features (i.e., dimensionality). More specifically, we use four different complexity scenarios that differ in the sample size and the dimensionality. The first scenario is the low-dimension low-sample size (LDLSS) scenario which is characterized by 5 features and 50 observations. The LDLSS scenario is typical for empirical investigations in which only a limited number of participants is involved and only a few features are collected (e.g., laboratory experiments). The second scenario is the low-dimension high-sample size (LDHSS) scenario which is described by 5 features and 50,000 observations and typical for analyzing observable numeric data (e.g., financial data). Third, we define the high-dimension low-sample size (HDLSS) scenario with 200 features and 50 observations. HDLSS problems often occur in text mining (e.g., sentiment analysis). And fourth, we define a high-dimension high-sample size (HDHSS) scenario with 200 features and 50,000 observations. An HDHSS scenario is typically given, for example, when detecting tumors in microarray data.

While a high dimensionality might lead to the problem of data piling, especially when there is only a low sample size (Ahn and Marron, 2010), a high sample size raises performance problems (Li et al., 2007). On the other hand, a small sample size

can cause a sparse-data bias (Greenland et al., 2000). We compare several classifiers in these scenarios in order to gain a better understanding on the final effect of the sample size and dimensionality on the performance of classification methods.

2.4.2 Data Characteristics

Some methods make explicit assumptions on data characteristics. A linear discriminant analysis, for example, assumes normally distributed data. A deviation from an assumption might lead to a biased classification model and finally to inaccurate predictions. We systematically generate data with respect to five data characteristics that are likely to affect the performance of the selected classification methods. The five data characteristics are described in the following.

2.4.2.1 Relation between Dependent and Independent Variables

The multivariate relationship between the dependent variable and independent variables might influence the performance especially of those classification methods that explicitly model the relationship between the class variable and the features (Kiang, 2003). We use two levels for this data characteristic: a linear relation and a quadratic relation.

2.4.2.2 Distribution of Feature Data

Some classification methods assume normally distributed feature data (Kiang, 2003). In order to test to what extent the performance of a classification method relies on the normality assumption, we generate two types of data: normally distributed data with $\mu \in [1, 3]$ and $\sigma = 3$ and exponentially distributed data with a rate of $\lambda \in [1/3, 1]$. We use a constant standard deviation to generate normally distributed data, so that only one parameter is drawn from a uniform distribution (i.e., μ in case of a normal distribution and λ in case of an exponential distribution).

2.4.2.3 Modality of the Class Distribution

Data might follow a distribution with two or more peaks (modes). Because some classification methods are sensitive to multi-modal data (Kiang, 2003), we generate either unimodal data or bimodal data where class $Y = 0$ is distributed into two regions that are separated by observations from class $Y = 1$. There are just two regions in the case of unimodal data, one for class $Y = 0$ and one for class $Y = 1$.

2.4.2.4 Weighting of Feature Data

Not all available features might be meaningful for assigning observations to classes. Methods, such as regularized logistic regression, thus allow penalizing features with

low or no discriminatory power. We use two different methods to weigh the features in our experiment. First, all m features are equally weighted. And second, we use the rank-order centroid method (Barron and Barrett, 1996) to assign a different weight to each feature. A rank is assigned to each feature in the case of an unequal weighting such that the first feature $i = 1$ gets ranking position 1, the second feature $i = 2$ gets ranking position 2 and so forth. This ranking vector is then transformed into m weights β_i with the following equation (Barron and Barrett, 1996):

$$\beta_i = \frac{1}{m} \sum_{k=i}^{m} \frac{1}{k}, \qquad \text{with } i = 1, \ldots, m. \tag{2.1}$$

2.4.2.5 Class Balance

Classifiers are supervised learning methods and their performance mainly depends on the training set. In the case of an unbalanced training set, a common problem is that classification methods might assign all observations to the majority class (Dupret and Koda, 2001; Farquad and Bose, 2012). This will lead to a high accuracy, because supervised learning algorithms are designed to maximize overall accuracy. We investigate the performance of classification methods with a balanced training set and an unbalanced training set where 80% of all observations belong to the first and 20% of the observations belong to the second class.

2.4.3 Experimental Setting

We test the main effects of the five data characteristics in each complexity scenario leading to six datasets for each scenario. A summary of the datasets is presented in Table 2.2. To test the effect of each data characteristic in each scenario, we generate 100 times a dataset for each case described in Table 2.2 and in each complexity scenario leading to totally 2,400 experiments. Each dataset consists of either $n = 50$ or $n = 50,000$ data (low or high sample size) and each observation in the dataset is described by either $m = 5$ or $m = 200$ features (low or high dimensionality). Each feature is represented by a value in $[0, 1]$ either drawn from a normal distribution or an exponential distribution (see Section 2.4.2.2). The feature values X_{ij} for each observation j are squared in the case of a quadratic relationship (see Section 2.4.2.1), weighted and summed up over all features i to a value Z_j. The weights β_i are set to 1 in case of an equal weighting or are computed with the rank-ordered centroid method otherwise (see Section 2.4.2.4). A noise term $\epsilon_j \sim \mathcal{N}(0, 1)$ is furthermore added to Z_j. In the case of a balanced dataset an equal number of observations are generated for the class $Y = 0$ and the class $Y = 1$. More specifically, we set $Y_j = 0$ if Z_j is lower than the median of Z. Otherwise, we set $Y_j = 1$. In the unbalanced case, we set $Y_j = 0$ for all observations j with Z_j being in the 0.2-quantile of Z (see Section

2.4.2.5). All observations with Z_j being between the 25- and the 75-percentile are assigned to class $Y = 1$ whereas all other observations are assigned to class $Y = 0$ in the case of a multimodal distribution (see Section 2.4.2.3).

Table 2.2: Simulation Cases for Each Complexity Scenario

Case	Generation of X_j	Computation of Z_j	Computation of Y_j
Linear (L)	$X_{ij} \sim \mathcal{N}(\mu_i, \sigma_i)$	$Z_j = \sum_i^m X_{ij} + \epsilon_j$	$Y_j = \begin{cases} 0 & \text{if } Z_j < Median(Z) \\ 1 & \text{if } Z_j \geq Median(Z) \end{cases}$
Quadratic (Q)	$X_{ij} \sim \mathcal{N}(\mu_i, \sigma_i)$	$Z_j = \sum_i^m X_{ij}^2 + \epsilon_j$	$Y_j = \begin{cases} 0 & \text{if } Z_j < Median(Z) \\ 1 & \text{if } Z_j \geq Median(Z) \end{cases}$
Non-Normal (NN)	$X_{ij} \sim Exp(\lambda_i)$	$Z_j = \sum_i^m X_{ij} + \epsilon_j$	$Y_j = \begin{cases} 0 & \text{if } Z_j < Median(Z) \\ 1 & \text{if } Z_j \geq Median(Z) \end{cases}$
Multimodal (M)	$X_{ij} \sim \mathcal{N}(\mu_i, \sigma_i)$	$Z_j = \sum_i^m X_{ij} + \epsilon_j$	$Y_j = \begin{cases} 0 & \text{if } Z_j < Percent_{25}(Z) \text{ OR } Z_j > Percent_{75}(Z) \\ 1 & \text{if } Z_j \geq Percent_{25}(Z) \text{ AND } Z_j \leq Percent_{75}(Z) \end{cases}$
Unequal Weights (UW)	$X_{ij} \sim \mathcal{N}(\mu_i, \sigma_i)$	$Z_j = \sum_i^m \beta_i X_{ij} + \epsilon_j$	$Y_j = \begin{cases} 0 & \text{if } Z_j < Median(Z) \\ 1 & \text{if } Z_j \geq Median(Z) \end{cases}$
Unbalanced (UB)	$X_{ij} \sim \mathcal{N}(\mu_i, \sigma_i)$	$Z_j = \sum_i^m X_{ij} + \epsilon_j$	$Y_j = \begin{cases} 0 & \text{if } Z_j \in Quantile_{20}(Z) \\ 1 & \text{if } Z_j \notin Quantile_{20}(Z) \end{cases}$

$\mu_i \in [1, 3]; \sigma_i = 3; \lambda_i \in [1/3, 1]; \epsilon_j \sim \mathcal{N}(0, 1)$

We perform a 10-fold cross validation to train the classification methods. 45 out of the 50 observations are used for training in the low sample size scenarios whereas 45,000 out of the 50,000 observations are used in the high sample size scenarios. The remaining observations from the training dataset are used to estimate the performance for selecting the best parameter levels. The 10-fold cross validation is thus performed for each parameter level combination. A test dataset consisting of further independently generated 50 or 50,000 observations is finally used to assess the performance of the classification methods. The classification model with the best performing parameter levels is applied to the test dataset. Figure 2.1 illustrates the method evaluation process.

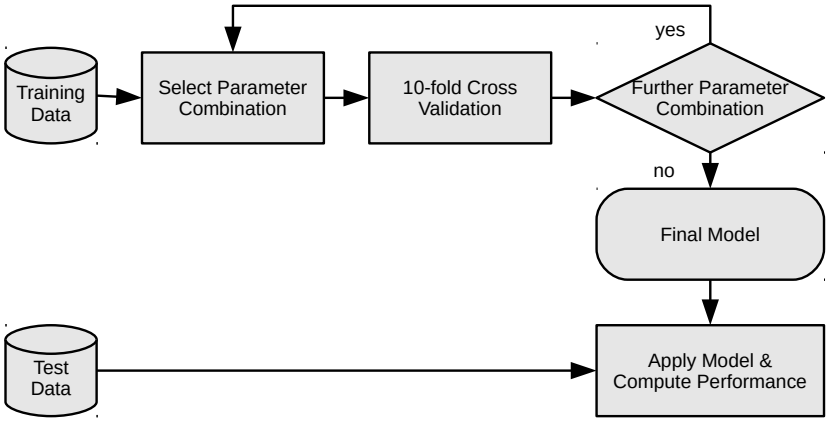

Figure 2.1: Classifier evaluation process

As described in Section 2.3, most classifier parameters significantly influence the methods' performance. The optimal values for these parameters are, however, rarely

known beforehand. We use the adaptive grid search introduced in (Kuhn, 2014) to find good parameter values. The adaptive grid search estimates the futility of parameter values just on a few re-samples with a generalized linear model and thereby avoids testing a full-factorial design of parameter values.

Our investigation thus consists of four complexity scenarios, six cases each testing one data characteristic, one hundred runs per case and complexity scenario and ten runs per classification method and parameter combination (10-fold cross validation). The number of tested parameter combinations is determined by the adaptive grid search approach. Classification methods requiring more than 250 seconds on average for classifying one dataset in a specific scenario are omitted in order to keep the computational effort actionable. With six cases and one hundred runs, the maximal average time per method and scenario is hence fixed to 150,000 seconds (= 41.667 hours).

We implemented the experiment on a 64-bit Windows Server 2012 machine with Intel Xeon E5-2690 processor and 128 GB RAM. We used only one core for training and prediction in order to keep the conditions constant for each classifier.

2.4.4 Performance Measures

We consider five performance measure for prediction accuracy: the percentage correctly classified (PCC), the area under the receiver operating characteristic (ROC) curve (AUC), the H-measure, the Brier Score (BS), and the F1 measure (F1). The five scores measure different performance dimensions. The PCC and F1 statistic measure the correctness of the predicted classes, the AUC and H-measure assess the discriminatory ability of a classification method and the BS measures the dimension probability prediction accuracy. We calculate these performance measures based on the vector of true o and predicted classes f.

We calculate the rank of each classification method in each complexity scenario and case and for each performance measure in order to compare the methods. The method with the best predictive performance gets rank one. We furthermore average the performance ranks to get a mean performance rank for each method. In most analyses we refer to this mean performance rank.

The time to train a classifier is used as another performance indicator. Training time is especially a crucial factor in real-time classification applications, such as real-time EEG-analysis (Müller et al., 2008) or real-time traffic sign detection (Greenhalgh and Mirmehdi, 2012). Thus, we also use the training time as another performance measure. Because classification methods are differently well prepared for a parallelized training stage, we will run our experiment only on one thread and do not make use of any parallelization opportunities.

2.5 Experimental Results

We present the results of our experiments in this section. We first discuss the benchmarking results and then the predictive performance of the tested classification methods. Second, we analyze and discuss the impact of the complexity scenarios and data characteristic cases on the predictive performance. We then analyze the correlations among the predictive performance measures. Finally, we discuss the results for the performance measure training time.

2.5.1 Benchmarking Results

We rely on the mean performance rank and present raw values in the online appendix [2]. As estimating a classification model can be very time-consuming, we did not evaluate the performance of all methods in all scenarios. Missing values in the following hence indicate that this classifier has not been evaluated in the focal scenario due to very high computational effort as discussed in Section 2.4.3. The high computational effort especially stems from a high number of parameter combinations as well as the fact that we evaluated each classifier on 100 datasets for each case and scenario. A summary of the results is presented in Table 2.3. The best performing classifier per complexity scenario and data case has a mean performance score of 1. Table 2.3 shows that the heterogeneous ensemble classifiers (SA_A, SA_3, and SA_5) show the best performance across all scenarios and datasets. This is in line with the findings of a comparison of classification methods for credit scoring (Lessmann et al., 2015). Regularized logistic regression (RLR), support vector machine with linear kernel (SVM_L), distance weighted discriminator with linear kernel (DWD_L), and boosted logistic regression (booLR) are in neither case among the top 5 performing classifiers rendering these methods not being recommendable for binary classification in several situations.

Interestingly, our results show that homogeneous ensemble classifiers (bCART, RF, booLR, and GBT) do not perform better than individual classifiers (except bagged CART in the LDHSS scenario). This underlines the necessity to build ensembles based on multiple classifiers, because at least one out of the four heterogeneous classifiers is among the best three methods in 22 out of the 24 cases and the top-3 simple average classifier is among the best three methods in 15 out of the 24 cases.

Simple averaging based on the top 3 performing classifiers is the best performing classification method in LDLSS and HDLSS scenarios. In LDHSS scenarios, we found bagged CART (bCART) to be the best classifier and the simple averaging based on all classifiers performed best in HDHSS scenarios. There is hence no classification method that always outperforms all other methods.

[2] Appendix can be found in Chapter 6

Table 2.3: Average classifier ranks for each case and scenario.

Method	*Low Dimensionality – Low Sample Size*						*Low Dimensionality – High Sample Size*						*High Dimensionality – Low Sample Size*						*High Dimensionality – High Sample Size*					
	L	**Q**	**NN**	**M**	**UW**	**UB**	**L**	**Q**	**NN**	**M**	**UW**	**UB**	**L**	**Q**	**NN**	**M**	**UW**	**UB**	**L**	**Q**	**NN**	**M**	**UW**	**UB**
LDA	6	22	6	24	6	16	10	14	2	14	7	4	21	24	15	23	6	9	2	7	2	6	5	8
RDA	11	6	5	6	12	15	–	–	–	–	–	–	22	10	11	10	14	19	–	–	–	–	–	–
LR	5	20	14	23	4	20	11	12	13	11	8	12	11	25	22	21	21	25	3	5	9	7	5	7
RLR	7	19	10	21	10	9	6	11	7	12	9	10	16	22	14	13	25	23	–	–	–	–	–	–
bayLR	4	18	13	25	5	10	9	13	12	13	6	13	19	16	23	24	23	16	6	6	8	8	3	6
kNN	18	8	15	10	23	8	13	4	9	3	13	14	8	**1**	5	7	10	10	–	–	–	–	–	–
NSN	14	21	9	22	14	6	8	15	3	15	11	3	10	14	10	20	16	6	8	8	6	9	7	**1**
SNN	15	11	18	11	15	17	–	–	–	–	–	–	9	2	3	6	7	3	–	–	–	–	–	–
NB	16	9	17	15	19	18	7	5	4	8	12	7	15	17	17	14	13	7	9	**1**	3	**1**	8	9
CART	25	25	24	7	22	25	15	10	15	10	15	5	7	15	7	8	3	24	7	9	4	3	9	**1**
C5.0	24	15	23	2	24	22	–	–	–	–	–	–	6	12	9	9	9	17	–	–	–	–	–	–
SVM_L	10	17	11	18	9	13	–	–	–	–	–	–	19	20	24	16	16	20	–	–	–	–	–	–
SVM_P	12	7	4	9	13	5	–	–	–	–	–	–	4	7	12	5	5	8	–	–	–	–	–	–
SVM_R	23	14	20	4	25	7	–	–	–	–	–	–	12	5	13	17	11	**1**	–	–	–	–	–	–
DWD_L	8	16	12	20	7	14	–	–	–	–	–	–	23	21	21	19	15	11	–	–	–	–	–	–
DWD_P	13	5	8	8	11	12	–	–	–	–	–	–	5	6	6	4	8	12	–	–	–	–	–	–
DWD_R	17	4	16	12	16	11	–	–	–	–	–	–	25	11	16	18	18	2	–	–	–	–	–	–
bCART	22	13	21	17	18	23	**1**	**1**	**1**	**1**	**1**	**1**	18	19	20	25	12	13	–	–	–	–	–	–
RF	20	12	19	14	17	19	–	–	–	–	–	–	13	13	18	11	20	5	–	–	–	–	–	–
booLR	21	24	25	19	20	24	14	9	14	6	14	15	24	23	19	15	24	22	–	–	–	–	–	–
GBT	19	23	22	16	21	21	12	7	11	5	10	8	14	18	25	22	22	15	–	–	–	–	–	–
SA_A	9	10	7	13	8	3	3	8	5	9	2	2	17	9	8	12	19	4	**1**	2	**1**	2	**1**	3
SA_3	**1**	**1**	**1**	**1**	**1**	**1**	2	2	10	2	3	11	**1**	4	**1**	**1**	**1**	21	4	4	7	4	2	5
SA_5	2	2	2	3	2	2	4	3	8	4	5	9	2	3	2	2	2	18	5	3	5	5	4	4
SA_7	3	3	3	5	3	4	5	6	6	7	4	6	3	8	4	3	4	14	–	–	–	–	–	–

Numbers indicate the mean performance rank. Bold values indicate the best performing classifier.
Missing values indicate that the method has not been evaluated for the particular scenario.

Distance weighted discrimination has been proposed as a method especially for the HDLSS scenario. This classifier, however, performs only outstanding in this scenario with a polynomial kernel. We – in contrast to Marron et al. (2007) – did not find evidence that the distance weighted discrimination with polynomial kernel always performs better in a HDLSS scenario than a support vector machine with a polynomial kernel. Only in the case of quadratic data, non-normal data or a multimodal class distribution has the distance weighted discrimination outperformed the support vector machine.

The stabilized nearest neighbor classifier (SNN) especially performed better than the k-nearest neighbor classifier (kNN) in the HDLSS scenario indicating that the stabilized variant improved the generalizability of the estimated models in this scenario. Sun et al. (2016) compared the stabilized nearest neighbor classifier to other nearest neighbor approaches only in scenarios with a low dimensionality. Our results demonstrate that this method is, however, especially recommendable for small training sets with high dimensionality.

The nearest shrunken neighbor classifier (NSN) is among the best performing classifiers in case of unbalanced data. This complements the findings by Tibshirani et al. (2003) who demonstrated that the nearest shrunken neighbor classifier has a higher predictive performance than a linear discriminant analysis (LDA) in a HDLSS

scenario. Our results also show that the nearest shrunken neighbor classifier outperforms the linear discriminant analysis in most cases in the HDLSS scenario. Additionally, we provide evidence that a nearest shrunken neighbor classifier is better than a linear discriminant analysis and most other methods (e.g., regularized discriminant analysis, logistic regression, k-nearest neighbor classifier and Naïve Bayes classifier) in the case of unbalanced training data.

2.5.2 Scenario and Case Results

To complement the analysis of our experimental results, we compare the performance of one classifier across all scenarios and cases in this subsection. The performance ranks presented in the previous section do not show differences in terms of the achievable performance in each scenario and case. Figure 2.2 shows the performance of the classifiers with the highest performance score per scenario and case. We selected the best classifier because it shows the highest possible performance among the investigated classification methods. The H-measure depicts the discriminatory ability of the classifiers.

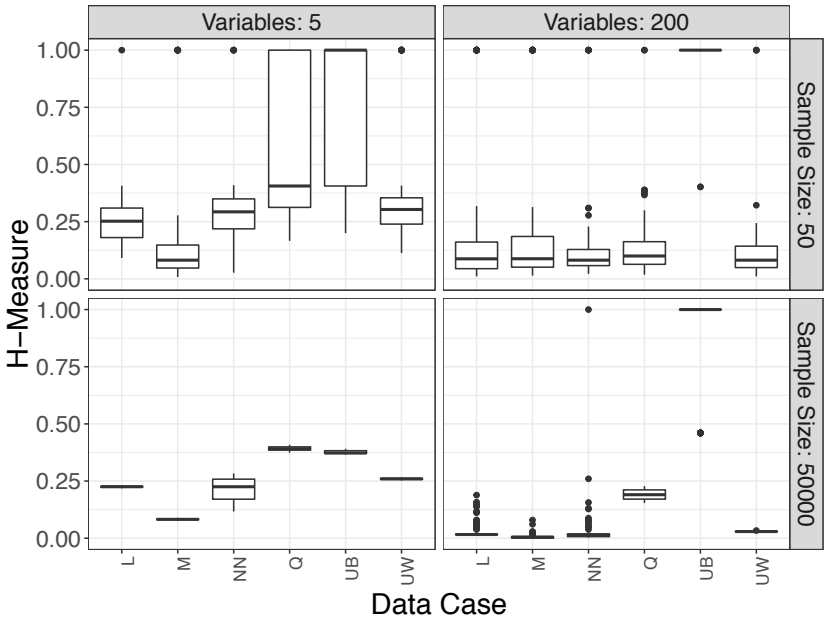

Figure 2.2: H-Measure of the Best Performing Classifier

Figure 2.2 shows that a higher dimensionality (i.e., higher number of variables) decreases the H-measure whereas a larger training set (i.e., larger sample size) decreases the variance of the classification performance. Taking the case with linear relation, normally distributed feature data, unimodal class distribution, equally important features and a balanced training set as reference, our results show that the classification performance mostly suffers from a change of the class distribution to

a multi-modal distribution. Interestingly, the H-measure is higher when an unbalanced dataset is used for training then when a balanced dataset is used. This is in contrast to the classification performance as indicated by the F1-measure (see Figure 2.3). The case with unbalanced data performs worst in terms of the F1-measure in all four complexity scenarios. Misclassifying the minority class is more serious than misclassifying the majority class in the case of unbalanced data. The H-measure, however, assumes that the misclassification costs are equal for each class leading to biased values in the case of unbalanced data (Thai-Nghe et al., 2011).

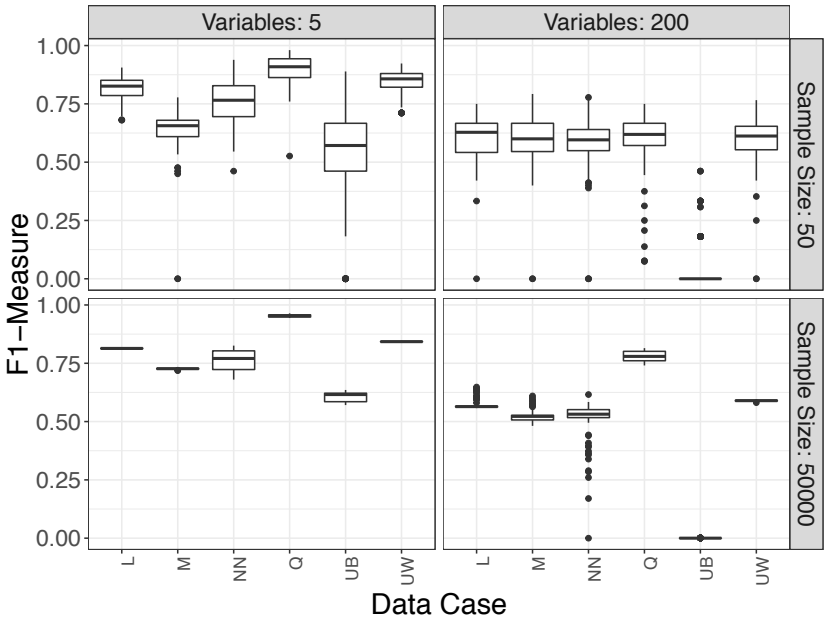

Figure 2.3: F1-Measure of the Best Performing Classifier

2.5.3 Correlation of the Performance Measures

We also analyzed the relations among our selected performance measures. Table 2.4 shows that AUC, H-measure, and Brier Score (BS) measured the same dimension of performance because they are highly correlated to each other. Another dimension of performance is measured by PCC and the F1-measure. This is in line with our definition of the performance measures (c.f. Section 2.4.4).

Table 2.4: Correlation of classifier across performance measures.

	AUC	PCC	H	BS	F1
AUC	1.000		–	–	–
PCC	0.459	1.000	–	–	–
H	0.999	0.462	1.000	–	–
BS	0.991	0.488	0.992	1.000	–
F1	0.267	0.795	0.267	0.275	1.000

The H-measure has been developed as a coherent alternative to the AUC measure (Hand, 2009). Our results indicate in line with the results by Lessmann et al. (2015) that the performance ranks generated based on the H-measure are very similar to those generated with the AUC measure. Both performance measures hence seem to be substitutable in empirical studies. In the case of binary classification problems, we recommend using one measure for each of the two dimensions – correctness of the predicted classes and discriminator ability because differences in terms of the performance ranks rather exists for indicators measuring different dimensions than for indicators measuring the same dimension.

2.5.4 Training Time

We trained the classification methods using only one thread. This allows analyzing the training time without parallel computing advances. As the methods are differently well suited for parallelization, a method being slower than another one on a single thread might outperform the other one when using multiple threads for training. However, the training time on a single thread divided by the number of possible threads provides a lower bound for the training time on a multi-threaded architecture. The results for the training time are depicted in Figure 2.4. Note that the training time for the simple average ensemble methods is the sum of the training time for all other methods. We thus excluded the simple average methods from the analysis of the training time.

Both complexity variables, the number of dimensions and the sample size, do have a significant influence on the training time. Only three methods required less than 100 seconds in the median in each scenario: logistic regression, Naïve Bayes, and the nearest shrunken neighbor. These methods are well suited for real-time classification (e.g., EEG classification). Figure 2.4 shows that the variance of the training time also significantly varies across the methods. Methods such as the linear discriminant analysis only show a low variance in the training time whereas other methods, such as the regularized logistic regression have a rather high variance.

We regressed the number of variables in the training set and the training sample size on the training time for those six methods that were included in each complexity scenario. The results of the OLS regressions are presented in Table 2.5. The ratio of the coefficient for the number of variables (dimensionality) and the coefficient for the sample size shows how sensitive the training time is on the dimensionality of the training data compared to the sample size. An increase of the number of variables compared to a commensurate increase in the sample size has the highest impact on the training time of the Naïve Bayes classifier. However, the absolute impact of the number of variables on the training time of a Naïve Bayes classifier is rather low as

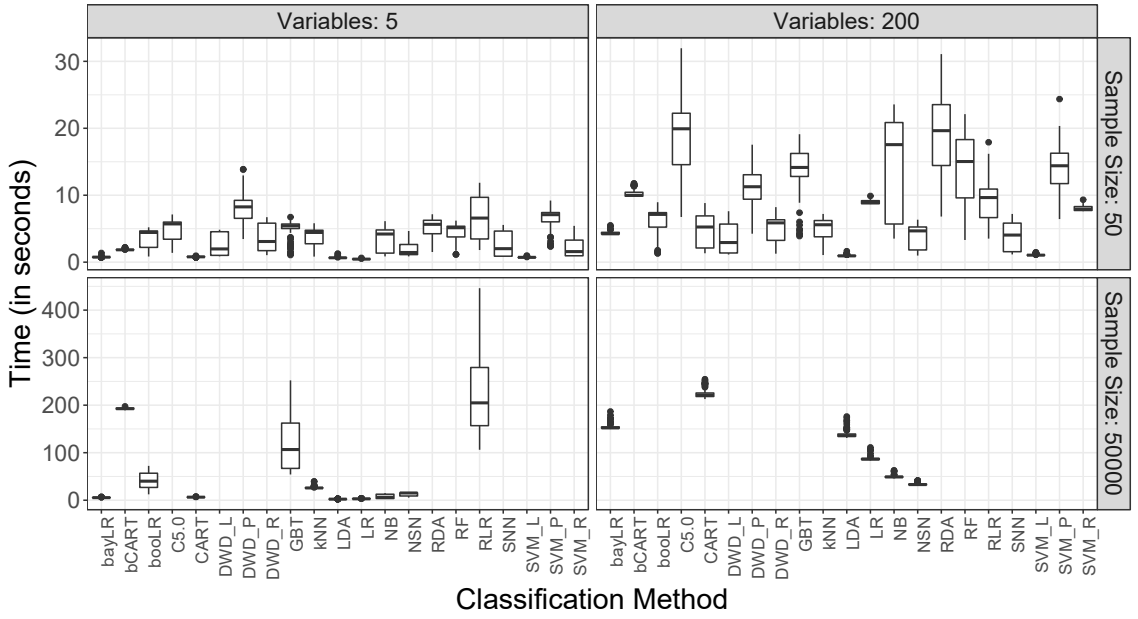

Figure 2.4: Training Time for each Classification Method in each Scenario

indicated by a rather small regression coefficient. Table 2.5 indicates that the nearest shrunken neighbor classifier is least sensitive to an increase of the dimensionality of the training data. This method is hence very well suited for real-time classification in high-dimensionality scenarios. For real-time classification of rather low-dimensional but large data we can especially recommend using the Naïve Bayes classifier or the nearest shrunken neighbor classifier.

Table 2.5: Influence of Dimensionality and Sample Size on Training Time.

Method	Intercept	# of Variables	Sample Size	R^2	# of Variables / Sample Size
bayLR	-37.24***	0.3877***	0.0007662***	0.682	506.00
CART	-34.91***	0.3648***	0.0007311***	0.481	498.97
LDA	-35.94***	0.3584***	0.0007127***	0.669	502.88
LR	-19.61***	0.2371***	0.0004042***	0.717	586.59
NB	-12.12***	0.1972***	0.0003185***	0.340	619.15
NSN	-9.464***	0.1253***	0.0003229***	0.415	388.05

*** $p < 0.001$

Table 2.5 also shows that the effect of the number of variables and the sample size on the training time is rather equal for the Bayesian logistic regression, CART and the linear discriminant analysis. Assuming that parallel computing is equally possible for all three methods, researchers and practitioners should make a decision between these three methods merely based on their classification accuracy.

2.6 Discussion

This paper investigates several classification methods and evaluates their performances in four complexity scenarios and six data characteristic cases. In contrast to almost all other existing comparative studies, we use a synthetic dataset in order to control the characteristics of the datasets on which we applied the classification methods. This comes with the big advantage that we can extract rules and guidelines for when to apply which classification method.

With controlled synthetic datasets we have shown that data characteristics significantly influence the performance of classification methods. The overall best performance was reached with a heterogeneous ensemble classifier. Homogeneous classifiers often did not show a significantly higher performance than individual classifiers. Our results, however, revealed that some classifiers are especially recommendable if the dataset to be classified has some certain characteristics. Logistic regression as an easy and often recommended and applied method (e.g., Kiang, 2003) showed rather a moderate performance and a worse performance than a simple average of the best performing classifiers in all cases and scenarios. Methods such as k-nearest neighbor classifier, support vector machines with polynomial kernel, distance weighted discriminator with polynomial kernel, bagged CART, C5.0 or Naïve Bayes classifier also performed better than a logistic regression in most investigated cases. We hence recommend training a large set of classifiers to finally build a heterogeneous ensemble, in general. In the case of a low data dimensionality and a high sample size, our results indicate using a bagged CART classifier rather than any other method. For the serious scenario of high dimensionality and only a small sample size, we can recommend using a heterogeneous ensemble, kernel-based classifiers with polynomial kernel, or stabilized nearest neighbor classifiers.

We assessed the performance of classification methods based on several performance measures. A correlation analysis of these measures showed that they measure two dimensions of performance: correctness of the predicted classes and discriminatory ability. We recommend that one measure for each dimension should be used to evaluate classifiers in empirical studies.

The case of unbalanced training data has been found to be the most serious case with respect to the F1-measure. In order to overcome a rather poor classification performance when the training set is unbalanced, researchers either can use resampling techniques to generate a balanced dataset or apply methods that are dedicated towards training on unbalanced data (López et al., 2014).

Our study is subject to three major limitations. First, we only investigated main effects of data characteristics on the performance of classifiers. Each case only differs in exactly one characteristic from the reference case. Because real-world datasets

typically are characterized by a mixture of the investigated data characteristics and often also by characteristics that have not been taken into account in this study, we encourage further research to also analyze interaction effects of the data characteristics and investigate the impact of further characteristics, such as a correlation between the feature data.

Second, we limited the set of investigated methods to 25 and hence did not include some existing classification methods like C4.5, quadratic discriminant analysis, or support vector machines with kernels other than linear, radial or polynomial. Furthermore, we did not consider classifiers that have been developed for specific classification methods. Our study, however, provides a generic experimental setting for comparing classification methods. This setting can be employed in future research to investigate further classifiers and data characteristics.

And third, we did not investigate the training time of the classification methods using parallelization techniques. We implemented the classifiers with a single-threaded model in order to keep the conditions constant for each classifier. Some classifiers can use several processing cores in parallel whereas this is not or only hardly possible for some other methods. Parallelization also depends on the concrete implementation of a classification method and thus makes it hard to compare the training time between the methods. Developing approaches for analyzing the training time using parallelization techniques provides an interesting avenue for further research.

References

Saeed Abu-Nimeh, Dario Nappa, Xinlei Wang, and Suku Nair. A comparison of machine learning techniques for phishing detection. In *2nd Annual eCrime Researchers Summit*, pages 60–69, 2007.

Jeongyoun Ahn and J.S. Marron. The maximal data piling direction for discrimination. *Biometrika*, 97(1):254–259, 2010.

Mark A. Aizerman, Emmanuel M. Braverman, and Lev I. Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837, 1964.

S. Aruna, S.P. Rajagopalan, and L.V. Nandakishore. An empirical comparison of supervised learning algorithms in disease detection. *International Journal of Information Technology Convergence and Services*, 1(4):81–92, 2011.

M. Asjad, A. Alam, and F. Hasan. A comparative study of classifier techniques for lift index data analysis. *Benchmarking*, 25(2):632–641, 2018.

B. Baesens, T. van Gestel, S. Viaene, M. Stepanova, J. Suykens, and J. Vanthienen. Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society*, 54(6):627–635, 2003.

F. Hutton Barron and Bruce E. Barrett. Decision quality using ranked attribute weights. *Management Science*, 42(11):1515–1523, 1996.

C. Baumgartner, C. Böhm, D. Baumgartner, G. Marini, K. Weinberger, B. Olgemöller, B. Liebl, and A.A. Roscher. Supervised machine learning techniques for the classification of metabolic disorders in newborns. *Bioinformatics*, 20(17): 2985–2996, 2004.

Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *5th Annual Workshop on Computational Learning Theory*, pages 144–152, 1992.

A. Bourouhou, A. Jilbab, C. Nacir, and A. Hammouch. Comparison of classification methods to detect the parkinson disease. In *2016 International Conference on Electrical and Information Technologies (ICEIT)*, pages 421–424, 2016.

Leo Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

Leo Breiman, Jerome Friedman, Richard A. Olshen, and Charles J. Stone. *Classification and Regression Trees*. Chapman & Hall/CRC, Boca Raton, 1984.

Jia Cai, Hongyan Wang, and Ding-Xuan Zhou. Gradient learning in a classification setting by gradient descent. *Journal of Approximation Theory*, 161(2):674–692, 2009.

Rich Caruana and Alexandru Niculescu-Mizil. An empirical comparison of supervised learning algorithms. In *23rd International Conference on Machine Learning*, pages 161–168, 2006.

Y.-C. Chen. A comparative assessment of classification methods for resonance frequency prediction of langevin piezoelectric transducers. *Applied Mathematical Modelling*, 35(7):3334–3344, 2011.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

T.M. Cover and P.E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967.

Nadia F.F. da Silva, Eduardo R. Hruschka, and Estevam R. Hruschka Jr. Tweet sentiment analysis with classifier ensembles. *Decision Support Systems*, 66:170–179, 2014.

Resul Das and Abdulkadir Sengur. Evaluation of ensemble methods for diagnosing of valvular heart disease. *Expert Systems with Applications*, 37(7):5110–5115, 2010.

Javier de Andrés, Manuel Landajo, and Pedro Lorca. Forecasting business profitability by using classification techniques: A comparative analysis based on a spanish case. *European Journal of Operational Research*, 30(1):68–76, 2005.

Dursun Delen. A comparative analysis of machine learning techniques for student retention management. *Decision Support Systems*, 49(4):498–506, 2010.

Thomas G. Dietterich. Ensemble methods in machine learning. In *International Workshop on Multiple Classifier Systems*, pages 1–15, 2000.

Stephan Dreiseitl, Lucila Ohno-Machado, Harald Kittler, Staal Vinterbo, Holger Billhardt, and Michael Binder. A comparison of machine learning methods for the diagnosis of pigmented skin lesions. *Journal of Biomedical Informatics*, 34(1): 28–36, 2001.

Philippe du Jardin. A two-stage classification technique for bankruptcy prediction. *European Journal of Operational Research*, 254:236–252, 2016.

Philippe du Jardin. Failure pattern-based ensembles applied to bankruptcy forecasting. *Decision Support Systems*, 107:64–77, 2018.

Georges Dupret and Masato Koda. Bootstrap re-sampling for unbalanced data in supervised learning. *European Journal of Operational Research*, 134(1):141–156, 2001.

Frank Eibe, Yong Wang, Stuart Inglis, Geoffrey Holmes, and Ian H. Witten. Using model trees for classification. *Machine Learning*, 32:63–76, 1998.

Reza Entezari-Maleki, Arash Rezaei, and Behrouz Minaei. Comparison of classification methods based on the type of attributs and sample size. *Journal of Convergence Information Technology*, 4(3):94–102, 2009.

M.A.H. Farquad and Indranil Bose. Preprocessing unbalanced data using support vector machine. *Decision Support Systems*, 53(1):226–233, 2012.

David Feldman and Shulamith Gross. Mortgage default: Classification trees analysis. *The Journal of Real Estate Finance and Economics*, 30(4):369–396, 2005.

Steven Finlay. Multiple classifier architectures and their application to credit risk assessment. *European Journal of Operational Research*, 210:368–378, 2011.

Ronald A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188, 1936.

Trevor Fitzpatrick and Christophe Mues. An empirical comparison of classification algorithms for mortgage default prediction : evidence from a distressed mortgage market. *European Journal of Operational Research*, 249(2):427–439, 2016. ISSN 0377-2217.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 28(2):337–407, 2000.

Jerome H. Friedman. Regularized discriminant analysis. *Journal of the American Statistical Association*, 84(105):165–175, 1989.

Jerome H. Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378, 2002.

Andrew Gelman, Aleks Jakulin, Maria Grazia Pittau, and Yu-Sung Su. A weakly informative default prior distribution for logistic and other regression models. *The Annals of Applied Statistics*, 2(4):1360–1383, 2008.

Jack Greenhalgh and Majid Mirmehdi. Real-time detection and recognition of road traffic signs. *IEEE Transactions on Intelligent Transportation Systems*, 13(4): 1498–1506, 2012.

Sander Greenland, Judith A. Schwartzbaum, and William D. Finkle. Problems due to small samples and sparse data in conditional logistic regression analysis. *American Journal of Epidemiology*, 151(5):531–539, 2000.

David J. Hand. Measuring classifier performance: a coherent alternative to the area under the ROC curve. *Machine Learning*, 77(1):103–123, 2009.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer, 2nd edition, 2008.

Chi Jin and Liwei Wang. Dimensionality dependent pac-bayes margin bound. In *Neural Information Processing Systems 2012*, 2012.

Melody Y. Kiang. A comparative assessment of classification methods. *Decision Support Systems*, 35(4):441–454, 2003.

Soo Y. Kim and Arun Upneja. Predicting restaurant financial distress using decision tree and AdaBoosted decision tree models. *Economic Modelling*, 36:354–362, 2014.

C. G. Kiran, L. V. Prabhu, V. Abdu Rahiman, and K. Rajeev. Traffic sign detection and pattern recognition using support vector machines. In *Seventh International Conference on Advances in Pattern Recognition*, pages 87–90, 2009.

Jochen Kruppa, Alexandra Schwarz, Gerhard Arminger, and Andreas Ziegler. Consumer credit risk: Individual probability estimates using machine learning. *Expert Systems with Applications*, 40(13):5125–5131, 2013.

Max Kuhn. Futility analysis in the cross-validation of machine learning models. *CoRR*, abs/1405.6974, 2014.

Salim Lahmiri. Features selection, data mining and financial risk classification: a comparative study. *Intelligent Systems in Accounting, Finance and Management*, 23(4):265–275, 2016.

Stefan Lessmann, Bart Baesens, Hsin-Vonn Seow, and Lyn C. Thomas. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, 247(1):124–136, 2015.

Hui Li, Young-Chan Lee, Yan-Chun Zhou, and Jie Sun. The random subspace binary logit (RSBL) model for bankruptcy prediction. *Knowledge-Based Systems*, 24(8): 1380–1388, 2011.

Y.G. Li, W.D. Zhang, and G.L. Wang. Prune support vector machines by an iterative process. *International Journal of Computers and Applications*, 29(2):164–169, 2007.

Victoria López, Alberto Fernández, and Francisco Herrera. On the importance of the validation technique for classification with imbalanced datasets: Addressing covariate shift when data is skewed. *Information Sciences*, 257:1–13, 2014.

Sebastian Maldonado, Christian Bravo, Julio López, and Juan Pérez. Integrated framework for profit-based feature selection and SVM classification in credit scoring. *Decision Support Systems*, 104:113–121, 2017.

A.I. Marqués, V. Garcia, and J.S. Sánchez. Two-level classifier ensembles for credit risk assessment. *Expert Systems with Applications*, 39(12):10916–10922, 2012.

J.S. Marron, Michael J. Todd, and Jeongyoun Ahn. Distance-weighted discrimination. *Journal of the American Statistical Association*, 102(480):1267–1271, 2007.

I. M. D. Maysanjaya, H. A. Nugroho, and N. A. Setiawan. A comparison of classification methods on diagnosis of thyroid diseases. In *2015 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, pages 89–92, 2015.

Mohammad R. Mohebian, Hamid R. Marateb, Marjan Mansourian, Miguel Angel Ma nanas, and Fariborz Mokarian. A hybrid computer-aided-diagnosis system for prediction of breast cancer recurrence (hpbcr) using optimized ensemble learning. *Computational and Structural Biotechnology Journal*, 15:75–85, 2017.

Klaus-Robert Müller, Michael Tangermann, Guido Dornhege, Matthias Krauledat, Gabriel Curio, and Benjamin Blankertz. Machine learning for real-time single-trial EEG-analysis: From brain-computer interfacing to mental state monitoring. *Journal of Neuroscience Methods*, 167(1):82–90, 2008.

Andrew Y. Ng. Feature selection, l_1 vs. l_2 regularization, and rotational invariance. In *21st International Conference on Machine Learning*, pages 78–85, 2004.

Suhail M. Odeh and Abdel Karim Mohamed Baareh. A comparison of classification methods as diagnostic system: A case study on skin lesions. *Computer Methods and Programs in Biomedicine*, 137:311–319, 2016.

David L. Olson, Dursun Delen, and Yanyan Meng. Comparative analysis of data mining methods for bankruptcy prediction. *Decision Support Systems*, 52(2):464–473, 2012.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *ACL Conference on Empirical Methods in Natural Language Processing*, pages 79–86, 2002.

Junghun Park, Hsiao rong Tyan, and C. c. Jay Kuo. Internet traffic classification for scalable qos provision. In *IEEE International Conference on Multimedia and Expo*, pages 1221–1224, 2006.

Arturo López Pineda, Ye Ye, Shyam Visweswaran, Gregory F. Cooper, Michael M. Wagner, and Fuchiang Tsui. Comparison of machine learning classifiers for influenza detection from emergency department free-text reports. *Journal of Biomedical Informatics*, 58:60–69, 2015.

Xingye Qiao, Hao Helen Zhang, Yufeng Liu, Michael J. Todd, and J.S. Marron. Weighted distance weighted discrimination and its asymptotic properties. *Journal of the American Statistical Association*, 105(489):401–414, 2010.

John Ross Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.

John Ross Quinlan. *C4.5 – Programs for Machine Learning*. Morgan Kaufmann, San Francisco, 1993.

Pradeep Ravikumar, Martin J. Wainwright, and John D. Lafferty. High dimensional ising model selection using l_1-regularized logistic regression. *The Annals of Statistics*, 38(3):1287–1319, 2010.

Paul Sajda. Machine learning for detection and diagnosis of disease. *Annual Review of Biomedical Engineering*, 8(1):537–565, 2006.

J. Shreve, H. Schneider, and O. Soysal. A methodology for comparing classification methods through the assessment of model stability and validity in variable selection. *Decision Support Systems*, 52(1):247–257, 2011.

J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural Networks*, 32: 323–332, 2012.

Ying Su, Yong Zhang, Donghong Ji, Yibing Wang, and Hongmiao Wu. Ensemble learning for sentiment classification. In *13th Workshop on Chinese Lexical Semantics*, pages 84–93, 2012.

Will Wei Sun, Xingye Qiao, and Guang Cheng. Stabilized nearest neighbor classifier and its statistical properties. *Journal of the American Statistical Association*, 111 (515):1254–1265, 2016.

Songbo Tan and Jin Zhang. An empirical study of sentiment analysis for chinese documents. *Expert Systems with Applications*, 34(4):2622–2629, 2008.

Nguyen Thai-Nghe, Zeno Gantner, and Lars Schmidt-Thieme. A new evaluation measure for learning from imbalanced data. In *The 2011 International Joint Conference on Neural Networks*, pages 537–542, 2011.

Robert Tibshirani, Trevor Hastie, Balasubramanian Narasimhan, and Gilbert Chu. Class prediction by nearest shrunken centroids, with applications to dna microarrays. *Statistical Science*, 18(1):104–117, 2003.

Tony Van Gestel, Bart Baesens, Johan A.K. Suykens, Dirk Van den Poel, Dirk-Emma Baestaens, and Marleen Willekens. Bayesian kernel based classification for financial distress detection. *European Journal of Operational Research*, 172(3): 979–1003, 2006.

Yun Wan and Qigang Gao. An ensemble sentiment classification system of twitter data for airline services analysis. In *15th IEEE International Conference on Data Mining Workshops*, pages 1318–1325, 2015.

Gang Wang, Jian Ma, and Shanlin Yang. An improved boosting based on feature selection for corporate bankruptcy prediction. *Expert Systems with Applications*, 41(5):2353–2361, 2014a.

Gang Wang, Jianshan Sun, Jian Ma, Kaiquan Xu, and Jibao Gu. Sentiment classification: The contribution of ensemble learning. *Decision Support Systems*, 57: 77–93, 2014b.

David West. Neural network credit scoring models. *Computers & Operations Research*, 27(11–12):1131–1152, 2000.

Matthew Whitehead and Larry Yaeger. Sentiment mining using ensemble classification models. In *2008 International Conference on Systems, Computing Sciences and Software Engineering*, pages 509–514, 2008.

Nigel Williams, Sebastian Zander, and Grenville Armitage. A preliminary performance comparison of five machine learning algorithms for practical ip traffic flow classification. *ACM SIGCOMM Computer Communication Review*, 36(5):7–15, 2006.

Rui Xia, Chengqing Zong, and Shoushan Li. Ensemble of feature sets and classification algorithms for sentiment classification. *Information Sciences*, 181(6): 1138–1152, 2011.

3 Feature detection in online customer reviews

Abstract

The numerous customer reviews available in online markets contain a variety of information, such as discussed product features, which are useful for participants in e-commerce. Product features must be automatically detected from customer reviews to utilize this information. In this paper, we demonstrate that the precision and recall of a bag-of-words model and a maximum entropy model applied to customer reviews to extract product features are contrary. Thus, we combine the two approaches to develop a highly accurate, novel ensemble approach to detect product features in online customer reviews.

Author: Tristan Wimmer

3.1 Introduction

The number of online customer reviews (OCRs) on e-commerce websites has increased substantially in recent years; thus, a huge amount of information is available for participants in e-commerce (Chen and Xu, 2017; Guo and Zhou, 2017). Consumers, retailers and manufacturers can benefit from the massive amount of information contained in OCRs (Yu et al., 2011; Jing et al., 2017).
Consumers consult OCRs during the purchase decision-making process to reduce uncertainty about whether a product fits their needs (Kang et al., 2017; Zhu and Zhang, 2010). This process in turn increases the purchase probability; thus, retailers can benefit from OCRs. (Akdeniz et al., 2013; Dorner et al., 2013). Apart from consumers and retailers, manufacturers can also utilize the information contained in OCRs. Manufacturers can identify the needs of (potential) consumers, improve their products and create marketing initiatives based on OCRs (Zhan et al., 2009).
In addition to these advantages, the massive number of OCRs has some drawbacks for consumers, retailers and manufacturers. During the decision-making process, consumers tend to process all available information they can obtain about a product of interest (Park and Lee, 2007). However, consumers have a limited processing capacity and become cognitively overloaded if there are too many customer reviews to process, resulting, inter alia, in cognitive strain – the so-called information overload phenomenon (Jacoby et al., 1974; Keller and Staelin, 1987; Malhotra, 1984). Retailers could provide consumers the opportunity to filter often-discussed product features to reduce the risk of information overload. For this reason, and to harness the benefits of OCRs, such as making recommendations or product improvements, retailers and manufacturers need to know which product features are discussed in OCRs. Due to the large quantity of OCRs, manual detection of product features is impractical. The automatic detection of product features from online customer reviews is essential to achieve the outlined benefits and to overcome the problems arising from the substantial quantity of OCRs.
However, the automatic detection of product features in OCRs is not a trivial task for several reasons. First, the writing style varies among consumers and across domains (e.g., different product categories, types of products), which complicates the application of rule-based approaches. A rule-based approach may be applicable for a highly focused application of feature detection, but a set of rules must be defined for each domain in order to apply a rule-based approach. Thus, rule-based approaches are impractical due to the variation in products, writing styles and domains.

Second, customers use different words to refer to the same feature (e.g., screen, monitor, display), which makes it difficult to apply a thesaurus. Third, OCRs are written in different languages, which creates challenges related to different grammar, vocabulary and writing style.

The existing approaches counter these problems by, for example, including external dependencies to detect product features in OCRs. Popescu and Etzioni (Popescu and Etzioni, 2005) use web search engines as external information resources. In practical applications, access to such external dependencies is difficult to obtain. Furthermore, the existing approaches also use opinion words (words describing a consumer's opinion about a product feature) to detect product features. These approaches search for product features located around an opinion word, such as the approach of Hu and Liu (Hu and Liu, 2004a). Thus, product features that are not associated with an opinion word are not detected. In practical applications, consumers, retailers and manufacturers are also interested in product features that are not associated with an opinion word, e.g., to identify product features that customers are missing. Such features are not necessarily associated with an opinion word.
In this study, we aim to develop an approach to automatically detect product features from OCRs, which fulfills the following conditions. First, the approach must be able to detect as many product features as possible, independently of whether there is an associated opinion word, to ensure a wide range of application. Second, our approach should neglect external dependencies and use only the OCRs themselves as data because external dependencies are expensive and difficult to maintain. Third, the approach should be able to work on a relatively small dataset. Most e-commerce platforms have hundreds, rather than thousands, of customer reviews. Further, recent machine learning algorithms, such as deep neural networks, typically require a considerable number of customer reviews to achieve high accuracy. New and small or medium-sized e-commerce platforms often have a limited number of customer reviews per product category and cannot provide the amount of data required to train these approaches with reasonable accuracy. To facilitate the application of feature detection by these platforms, our approach should work well with only hundreds of customer reviews. Fourth, our method should be customizable to different languages because e-commerce platforms typically have separate shops for different countries and, hence, contain customer reviews in different languages.
Somprasertsri and Lalitrojwong (Somprasertsri and Lalitrojwong, 2008) counter these problems with a promising maximum entropy (ME) model. ME models are among the most popular methods for information extraction (Zhang et al., 2010). Somprasertsri and Lalitrojwong's proposed model achieves high accuracy and precision but relatively low recall. Another common method for text classification is bag-of-words (BOW) models (Bloehdorn and Hotho, 2006; G. Forman, 2003). BOW models are often used for opinion mining, which is frequently performed simultaneously with product feature detection (Yu et al., 2017; Mohey, 2016; Khan et al., 2014; Whitelaw et al., 2005; Socher et al., 2013). Intuitively, if all words of a customer review are contained in the bag, the model would achieve a recall of one, whereas

the precision would be low. We compare a BOW model to a ME model and find that the accuracy of the BOW model is high and that its precision and recall are opposite those of the ME model. Thus, a combination of the two approaches can improve the detection accuracy. Thus far, existing research has not applied a BOW model for feature detection in customer reviews nor combined a ME model and a BOW model, even though the precision and recall of the methods are opposite and would satisfy our conditions.

In this paper, we propose a highly accurate ensemble approach that combines a BOW model with a ME model to detect product features in online customer reviews. Furthermore, our proposed approach works independently from the presence of any opinion words, ensuring that it detects as many product features as possible. In addition, our approach does not use external dependencies to ensure high applicability. Furthermore, our proposed approach achieves high accuracy even when only a few customer reviews are available for training.

The remainder of this paper is organized as follows. In the next section, we provide an overview of related work. Then, we present our proposed approach in section 3.4. In section 3.5, we present some benchmarks and describe the datasets used for the empirical evaluation. The results are outlined in section 3.6. We validate our results in section 3.7 and conclude this paper with a summary of the results and a discussion.

3.2 Related work

The existing research on feature detection can be broadly classified into two streams based on the objective. The first stream follows the initial work of Hu and Liu (Hu and Liu, 2004a) and pursues detection of product features from customer reviews by using opinion words or aiming to determine the sentiment of a product feature (Hu and Liu, 2004a,b; Xu et al., 2010; Qiu et al., 2009; Zhuang et al., 2006; Kobayashi et al., 2007; Jakob and Gurevych, 2010; Stoyanov and Cardie, 2008; Wang and Wang, 2008). Therefore, this stream focuses on the detection of product features that appear together with one or more opinion words. Product features that are not described by an opinion word are neglected. This process differs from the second stream of research, which aims to detect product features regardless of opinion words. This differentiation largely determines the application possibilities. For example, a retailer who wants to give customers the opportunity to consider the most-discussed product features needs to detect all product features. Given the following OCR from the dataset used in (Hu and Liu, 2004a), the approaches of the first stream do not label *"zoom"* as a product feature because it is not described by an opinion word, whereas *"closeup mode"* and *"battery"* are annotated as product features because they have associated opinion words.

"the same 4mp chip from the 4500 camera, plus a 3x zoom with the ability to expand upon that with extenders, great closeup mode, long lasting rechargable battery, etc etc."

Thus, if the focus is on the determination of a product feature's sentiment, the first stream is suitable. However, if all features, independently of co-occurrence with an opinion word, are needed, the second stream is preferred. In this paper, we aim to extract as many product features as possible, irrespective of associated opinion words. Thus, we focus on the second stream of literature.
Popescu and Etzioni (Etzioni et al., 2005; Popescu and Etzioni, 2005) investigate the detection of product features independently of opinion words. Their algorithm determines whether a noun or noun phrase is a feature by computing the pointwise mutual information (PMI). They compute the PMI score between a given phrase and a product-class-specific discriminator (e.g., TVs). Access to web search engines is required to calculate the PMI score, but access to web search engines for commercial use is not free of charge and thus limits the applicability, especially for small-sized retailers.

Somprasertsri and Lalitrojwong (Somprasertsri and Lalitrojwong, 2008) neglect external dependencies. They focus on the extraction of product features from OCRs and exclude the detection of opinion words, as well as their polarity. Somprasertsri and Lalitrojwong use a two-step approach. First, they extract product features with a classifier; second, they search for additional product features that consist of more than one word (e.g., mouse wheel). In the first step, they use a ME model classifier with a set of attributes to detect product features. The underlying idea of an ME model is that its attributes are conditionally dependent of each other. ME selects the model with the highest entropy as the model with the probability distribution that best represents the current data (Somprasertsri and Lalitrojwong, 2008; Ratnaparkhi, 1998). They compare different ME models based on different sets of attributes. The ME model with the highest F1 is based on three attributes. First, Somprasertsri and Lalitrojwong use part-of-speech (POS) tags (Somprasertsri and Lalitrojwong, 2008) as attribute. Further they assume that infrequent words are not likely to be product features. If a word in the training set occurs fewer than five times, it is classified as rare. This information is passed to the ME as the second attribute. Third, they use the context of each word: the four words before and after a word are used as context information. In the second step, Somprasertsri and Lalitrojwong (Somprasertsri and Lalitrojwong, 2008) apply a natural language processing technique to consider product features that consist of more than one word. The word is annotated as a feature if its head noun matches the product features

extracted by the ME. We focus on product features consisting of one word; thus, only the first step is applicable in our study. They demonstrate that their proposed ME model achieves high accuracy but relatively low recall.
We contribute to the existing research by combining an ME model, which is one of the most commonly used methods in information extraction (Zhang et al., 2010), and a BOW model into a novel ensemble approach to detect as many product features as possible and to counteract the low recall of ME models demonstrated by Somprasertsri and Lalitrojwong (Somprasertsri and Lalitrojwong, 2008).

3.3 Review model

An OCR may consist of product features that represent a product characteristic (e.g., the *screen size* of a TV). We follow the existing literature and limit features to be nouns because nouns are considered to be feature candidates (Archak et al., 2011). Product features can be explicitly or implicitly mentioned in customer reviews. For example, in the customer review *"The camera can easily put into the pocket."*, the customer writes about the size without mentioning it explicitly. Thus, the size is an implicit product feature. Explicit product features are mentioned directly by the customer (e.g., small camera size). In this paper, we follow Somprasertsri and Lalitrojwong (Somprasertsri and Lalitrojwong, 2008) and focus on the detection of explicit product features in OCRs.
Formally, a dataset of OCRs consists of I customer reviews ($R = \{r_1, ...r_I\}$). These customer reviews are split into a training set and a testing set. The training set consists of all reviews r_i with $i \in [1, z]$, and the testing set consists of reviews with $i \in [z+1, I]$. Each review r_i may contain N sentences $r_i = \{s_{i,1}, ..., s_{i,N}\}$. Each sentence may have M sequences $s_{i,n} = \{q_{i,n,1}, ..., q_{i,n,M}\}$ defined as a part of a sentence delimited by punctuation, such as ",". Each sequence $q_{i,n,m}$ consists of P words $q_{i,n,m,p} = \{w_{i,n,m,1}, ..., w_{i,n,m,P}\}$. Each word is represented by a POS tag $p(w_{i,n,m,p}) = ps$ *with* $ps \in \{$*UH, LS, DT, NNS, VBD, JJ, NN, CC, RB, IN, PRP, VBN, NNP, JJS, VBP, VB, PRP, VBG, VBZ, MD, RP, TO, EX, WRB, CD, WDT, JJR, RBS, RBR, PDT, WP, NNPS, FW*$\}$. Further, each sequence may contain D features $q_{i,n,m,p} = \{f_{i,n,m,p,1}, ..., f_{i,n,m,p,D}\}$. We limit the input to the OCR itself and its POS tags. POS tags are appropriate because they can be easily determined by a POS tagger, such as the Stanford Log-linear Part-Of-Speech Tagger[1]. Furthermore, POS taggers are available in almost every language and thus can be easily adapted for OCRs written in different languages.

[1] https://nlp.stanford.edu/software/tagger.shtml

3.4 Proposed approach

Feature detection in online customer reviews is a kind of text classification because the aims is to classify whether a word is a product feature. BOW models are a common method for text classification (Bloehdorn and Hotho, 2006; G. Forman, 2003). A BOW model for feature detection matches each word of a new customer review with the words contained in the bag. If the word is an element of the bag, the word is annotated as a product feature. Thus, as many product features as possible should be included in the bag to detect a high proportion of available product features. If all possible product features are contained in the bag, the BOW model would achieve a recall of one. Intuitively, customers write about a limited number of product features. Thus, only a relatively small number of customer reviews must be processed to obtain many possible product features. However, product features that are rarely discussed have a lower probability of being contained in the training set and thus in the bag. The BOW approach does not consider any contextual information, which is sometimes indispensable in determining whether a word is a product feature because a specific word is not necessarily a product feature every time a customer uses it. For example, whether the word *restaurant* is a product feature depends on the context. If a customer of a hotel writes about a restaurant in town, the restaurant is not a product feature because the hotel has no influence on the restaurant. By contrast, if the customer writes about the hotel restaurant, it is definitely a product feature. A possible solution to this problem is to determine a threshold. This threshold can be defined as how often a word occurs as a feature. If a word occurs several times but is rarely a feature, the word is not added to the bag. However, in this scenario, rare features are not identified by the BOW model. The approach proposed by Somprasertsri and Lalitrojwong (Somprasertsri and Lalitrojwong, 2008) may be compliant with a BOW model as follows. As outlined in section 3.2, they propose an ME model to detect product features. In contrast to BOW models, the ME model proposed by Somprasertsri and Lalitrojwong (Somprasertsri and Lalitrojwong, 2008) considers the context information for classification. Because the ME model uses additional information for classification, it might be suitable for detecting further product features. In our preliminary investigation, we compared the two approaches, i.e., BOW and ME, and found that the recall and precision show opposite trends. The ME model in (Somprasertsri and Lalitrojwong, 2008) achieves high precision but low recall, whereas the BOW model achieves high recall and low precision. Therefore, the ME model classifies fewer words as product features but the detected words are true product features, whereas the BOW model detects more product features but also incorrectly predicts more words to be features. We propose a new ensemble approach that combines a BOW model and an ME model to take advantage of the benefits of both models. We combine the results of both ap-

proaches to build our ensemble model. We define the BOW model to be the leading approach because the words in the bag are definitely feature candidates. Furthermore, our preliminary investigation demonstrated that the BOW model achieves a significantly higher recall, and thus detects more product features, than does the ME model. The ME model can reverse the prediction of the BOW model in one scenario. If the BOW approach classifies a word as a non-feature (because it is not in the bag) but the ME model predicts it as product feature, the classification result of the ME model is preferred because the ME model is able to detect features that are not contained in the bag. Table 3.1 depicts the decision matrix of our ensemble model.

Table 3.1: Decision Matrix

Predicted by BOW \ Predicted by ME	Feature	Non-feature
Feature	Feature	Feature
Non-feature	**Feature**	Non-feature

3.5 Experimental investigation

In this section, we introduce some benchmarks and different variants of our proposed ensemble approach. Further, we describe the datasets used to evaluate our ensemble approach and further benchmarks.

3.5.1 Algorithms

In this section, we present our proposed approach and benchmarks for comparison. The first benchmark is a naive approach (NP), which is followed by three hidden Markov chain models (HMM_1, HMM_2, HMM_3). Next, we apply two BOW models (BOW_1, BOW_2). Furthermore, we apply ME models in two forms (ME_1, ME_2). Last, we introduce four forms of our proposed ensemble model based on the BOW and ME models (EM_1, EM_2, EM_3, EM_4). Table 3.2 gives an overview of the applied algorithms.

Table 3.2: Algorithm Overview

Abbreviation	Algorithm	Description
NP	–	All nouns are features
HMM_1	Hidden Markov Chain	Based on a single word
HMM_2	Hidden Markov Chain	HMM_1 with optimization
HMM_3	Hidden Markov Chain	Based on sequences
BOW_1	Bag of Words	BOW without pruning and without rare word condition
BOW_2	Bag of Words	BOW with pruning and without rare word condition
BOW_3	Bag of Words	BOW with pruning and with rare word condition
ME_1	Maximum Entropy	Trained with context and POS tags
ME_2	Maximum Entropy	Trained with context, POS tags and rare word condition
EM_1	Bag of Words+Maximum Entropy	BOW_2 + ME_1
EM_2	Bag of Words+Maximum Entropy	BOW_3 + ME_2
EM_3	Bag of Words+Maximum Entropy	BOW_2 + ME_2
EM_4	Bag of Words+Maximum Entropy	BOW_3 + ME_1

3.5.1.1 Naive approach

We implement a naive approach (NP) as a benchmark. As outlined in 3.3, we define features as nouns. Therefore, we formulate a simple baseline that predicts every word $w_{i,n,m,p}$ of the testing dataset as a feature if $p(w_{i,n,m,p}) \in \{NN, NNS, NNP, NNPS\}$. Every plural, singular, proper singular and proper plural noun is predicted to be a feature.

3.5.1.2 Hidden Markov chain model

Hidden Markov models (HMM) are popular for information extraction and include syntactic information (Zhang et al., 2010). Whether a word is a feature may depend on the syntax of the customer review; thus, we apply three different HMM models. In general, an HMM model is a doubly stochastic process consisting of a number of states and observations, transition probabilities, emission probabilities and an initial state distribution (Rabiner and Juang, 1986). The emission probabilities include the probability that a state emits into an observation, whereas the transition probabilities are the probabilities that a feature is followed by a further feature or non-feature and vice versa. In HMM_1, the POS tags $p(w_{i,n,m,p})$ of the currently considered word $w_{i,n,m,p}$ represent observations. HMM_1 may predict non-nouns as features. In accordance with the assumption that only nouns are feature candidates, HMM_2 optimizes the output of HMM_1 in a post-processing step. Therefore, HMM_2 annotates every feature candidate as a non-feature if $p(w_{i,n,m,p}) \notin \{NN, NNS, NNP, NNPS\}$. In HMM_1 and HMM_2, the states are *feature* and *non-feature*. The POS tag of the actual processed word may be insufficiently informative to detect product features in OCRs. Thus, whether a word is a feature may not depend on $p(w_{i,n,m,p})$ but on the sequence $q_{i,n,m}$ it belongs to. Therefore, we implement HMM_3, which takes sequences of POS tags into consideration. The observations of HMM_3 are the merged POS tags of each sequence $q_{i,n,m}$. The states are *(feature, non-feature)* based on each sequence. Thus, the outcome of HMM_3 is the prediction of whether a sequence is

a feature. According to our definition, sequences cannot be features; thus, we annotate each noun in a sequence as a feature if the sequence itself is predicted to be a feature.

3.5.1.3 Bag of words

In addition to the syntax, the word itself may be relevant to whether it is a feature. Intuitively, customers predominately write about the same product features because each service or product has a limited number of possible features. Thus, we assume that a BOW model might be suitable to detect product features from OCRs. In general, a BOW model is a very simple and specific type of vector space model that disregards grammar (Ngo-Ye and Sinha, 2014; Chan and Chong, 2017). First, all words from each customer review r_i in the training set that are annotated as features f_i $with$ $i \in [1, z]$ are selected. These words are stored in the BOW. After all customer reviews in the training set are processed, the BOW contains, independently of their occurrence frequency, all features of the training set, so that $\forall f_i : f_i \in BOW with\ i \in [1, z]$. In the next step, all customer reviews of the testing set are processed. Therefore, each customer review r_i in the testing set is split into its words. Each of these words and each of the words in the BOW are transformed into lower case and stemmed. Afterwards, each word is labeled as a feature candidate, which fulfills the condition $w_{i,n,m,p} \in BOW\ with\ i \in [z+1, I]$. Thus, all feature candidates in the testing set that have been manually identified as features in the training set are tagged.
We apply three types of BOW models. First, we apply a model as outlined before (BOW_1). The existing approaches of the first stream of literature, such as Hu and Liu (Hu and Liu, 2004a), search for product features located around an opinion word that describes these features. As outlined above, these approaches are limited to detecting only product features that are described by an opinion word. Nevertheless, in a preliminary investigation, we found that adjectives are often located near product features. These adjectives do not have to be opinion words nor do they necessarily describe the product feature. Thus, we introduce BOW_2, which splits each sentence $s_{i,n}$ $with$ $i \in [z+1, I]$ into M sequences. If there is no adjective in the sequence $q_{i,n,m}$, each detected feature candidate is annotated as a non-feature. Thus, BOW_2 might produce fewer false positives because product features without a nearby adjective are classified as non-features. In contrast to the approaches of the first stream of literature, BOW_2 is not limited to detecting only product features that are described by an opinion word because the referencing of the adjective to a special feature does not play any role. Somprasertsri and Lalitrojwong (Somprasertsri and Lalitrojwong, 2008) set a threshold of five for feature candidates; that is only words that occur five times or more can be feature candidates because some words

might be a feature in only a special context. BOW_3 implements this threshold and thus ensures that words that are frequently not product feature are excluded from the bag and, hence, are not detected.

3.5.1.4 Maximum entropy

BOW models detect product features based on the words themselves, whereas the three hidden Markov chains ignore the words and consider the syntax based on the POS tags. BOW models might tend to incorrectly detect words as feature candidates because depending on the context, the same word can be a feature or non-feature. The three HMMs predict feature candidates based on the order probability of the POS, which implies that the syntax around a product feature is fairly consistent. Somprasertsri and Lalitrojwong combined the syntax and the word itself via an ME model Somprasertsri and Lalitrojwong (2008). An ME model is a probabilistic approach that classifies data based on a predefined, weighted set of variables. Somprasertsri and Lalitrojwong Somprasertsri and Lalitrojwong (2008) used POS tags and the context of each word. Therefore, they passed the words and the POS tags of the four words before and after each target word to the ME model. Further, they assumed, that infrequent words are not likely to be product features. They defined a word as rare if it occurred in the training set fewer than five times. This information was also passed to the ME model. Table 3.3 summarizes the features used to train the best-fitting ME model.

Table 3.3: Maximum Entropy Features

Variable	Description
Context	All words in a [-4,+4] window around the considered word
POS Tag	All POS tags in a [-4,+4] window around the considered word
Rare Word Condition	Whether the word occurs fewer than 5 five times in the training set

We build two different ME models. To obtain results comparable to those of BOW_2 and BOW_3, we train ME_1 with the same features as those of BOW_2 (the context and POS tags). Furthermore, we train ME_2 with the rare word condition of BOW_3 to exclude the possibility that the threshold is dependent on the dataset. Thus, we use the same set of features as that of Somprasertsri and Lalitrojwong (Somprasertsri and Lalitrojwong, 2008). They used the Maxent toolkit version 2.4.0. We use the more recent Maxent toolkit version 3.0.0[2].

3.5.1.5 Ensemble approach

We build four ensemble models (EM_1, EM_2, EM_3, EM_4) to map all combinations of the presented BOW and ME models. EM_1 is a combination of BOW_2 and ME_1

[2]available at: `https://sourceforge.net/projects/maxent/files/Maxent/3.0.0/`

(both without the rare word condition), and EM_2 is a combination of BOW_3 and ME_2 (both with the rare word condition). EM_3 is a combination of BOW_3 and ME_1 (with and without the rare word condition), and EM_4 is a combination of BOW_2 and ME_2 (without and with the rare word condition).

3.5.2 Datasets

We investigate the performance of our ensemble approach with two datasets. First, we collect 400 English customer reviews for hotels as a service from a hotel booking platform. Second, we use 200 randomly selected English customer reviews for TVs (physical product) from the newegg.com dataset presented in (Ngo-Ye and Sinha, 2014). We manually annotate all product features in both datasets. In the hotel dataset, we neglect alleged features (e.g., restaurants) that may not be in the sphere of influence of the hotelier. The characteristics of both datasets are summarized in Table 3.4. The hotel reviews are half as long as the TV reviews, but the numbers of sentences are similar. The similarity between customer reviews within the datasets is low in both cases.[3] Thus, customers have a different writing styles. Furthermore, the numbers of discussed features differ. Hotel customers discuss 95 different features and TV customers 78 different features, whereby they use 1.48 and 1.33 product features per review on average. A total of 113 hotel customer reviews and 66 TV reviews do not contain any product features[4]. Table 3.4 presents an overview of both datasets.

Table 3.4: Dataset Overview

	Hotel Dataset	TV Dataset
Number of reviews	400	200
Mean review length in words (SD)	22.42 (19.45)	43.18 (57.47)
Mean number of sentences (SD)	2.04 (1.55)	2.95 (3.71)
Mean cosine similarity between reviews (SD)	0.16 (0.05)	0.14 (0.04)
Number of product features	594	266
Number unique product features	95	78
Percentage of unique product features	0.16	0.29
Number of reviews without features	113	66
Mean number of features per review (SD)	1.48 (1.36)	1.33 (1.23)
Number of product feature with occurrence <5	31	42
Percentage of product features with occurrence <5	0.33	0.64

[3] We calculated the cosine similarity between reviews based on the POS tags. POS tags are suitable for representing the uniform structure of sentences. The cosine similarity ranges between zero (very dissimilar) and one (equal). The cosine similarity of customers reviews for the hotel dataset is 0.16, and that of the TV dataset is 0.14.

[4] For example, the OCR *"I loved absolutely everything!"* doesn't contain any hotel features.

3.6 Experimental results

3.6.1 Results of the bag of words and maximum entropy Models

We applied 10-fold cross-validation to obtain robust results.

Table 3.5: Results for the Hotel Dataset for the BOW and ME Models

	F1	Precision	Recall
BOW_1	0.6696	0.5271	0.9255
BOW_2	0.7101	0.6087	0.8559
BOW_3	0.7471	0.7563	0.7390
ME_1	0.7303	0.8818	0.6260
ME_2	0.6952	0.8916	0.5741

Table 3.6: Results for the TV Dataset for the BOW and ME Models

	F1	Precision	Recall
BOW_1	0.5960	0.4626	0.8598
BOW_2	0.6394	0.5314	0.8152
BOW_3	0.6178	0.7334	0.5376
ME_1	0.5181	0.9091	0.3679
ME_2	0.5154	0.9154	0.3640

Tables 3.5 and 3.6 show the results for the BOW and ME models. Within the BOW approach, BOW_2 and BOW_3 perform best. BOW_2 classifies words as product features only if there is an adjective within the sequence, whereas BOW_3 extends BOW_2 by implementing a threshold. Words that occur fewer than five times are classified as non-features even if they are contained in the bag. Thus, words that are rarely product features and are more often non-features are classified as non-features, as outlined in section 3.4. Tables 3.5 and 3.6 show that the threshold improves the accuracy for the hotel dataset, whereas it decreases the accuracy for the TV dataset. Overall, the accuracy of the BOW models is fairly high, which confirms our assumption that BOW models are suitable for detecting product features in OCRs. As many product features as possible must be contained in the training set and thus in the bag for a BOW model to achieve high accuracy. Furthermore, the training set must be relatively small to avoid excessive effort annotating product features. Thus, customer reviews must be characterized by a high feature-review rate, i.e., a limited number of customer reviews must contain almost all product features, in order for a BOW model to detect many product features. Figure 3.1 represents the features gained by adding the features of an additional customer review. Initially, adding the product features from a customer review to the bag fills the bag with, on average,

approximately 1.2 new product features. The greater the number of features already in the bag is, the lower the probability that a feature to be added is not already in the bag. A strong negative slope is observed up to approximately 100 OCRs. Thus, accounting for features in additional reviews becomes less worthwhile because the probability of adding an unknown product feature to the bag is greatly reduced.

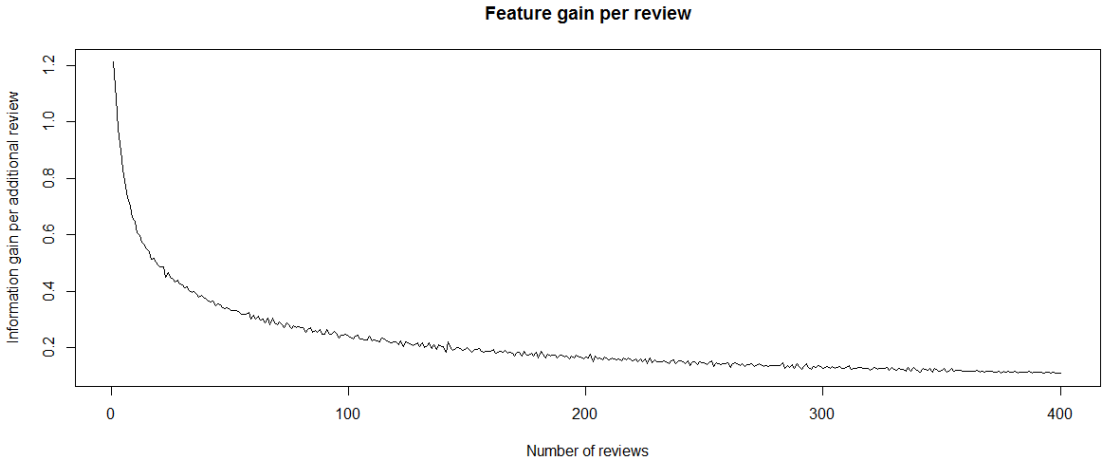

Figure 3.1: Feature information gain per review

Within the ME models, ME_1 performs best for both datasets. ME_1 is based on the context information and POS Tags, whereas ME_2 includes the rare word threshold. As outlined above, we adopt the threshold of five from Somprasertsri and Lalitrojwong (Somprasertsri and Lalitrojwong, 2008). We tested different thresholds and found that only a threshold of five improved the accuracy. When comparing the ME and BOW models, the trends in precision and recall are opposite. The BOW models have substantially higher recall but considerably lower precision than those of the ME models. Thus, the BOW models detect more product features than do the ME models, but they also misclassify more words as features. Words in the bag are not necessarily product features every time they are used by customers. The word *restaurant* is a feature if it belongs to the hotel, whereas when referring to a restaurant in town, it is not a product features. BOW models do not consider this differentiation, resulting in a lower recall. The ME models achieve high precision, indicating that they detect real product features and misclassify only a few words as features. The low recall of the ME models indicates a failure to detect some product features. However, in contrast to the BOW models, which are limited to detecting product features that are already in the bag, ME models are able to detect all product features independently if they have previously been observed in the training set. We investigated the results of the best models (BOW_3 and ME_1) for the hotel dataset. Table 3.6.1 represents the confusion matrix for the

hotel dataset. ME_1 has fewer false positives but also fewer true positives. The high number of false negatives indicates that ME_1 misses twice as many product features as BOW_3 does. Furthermore, 81.94% of the product features not detected by ME_1 are detected by BOW_3. Moreover, ME_1 detects an additional 21.05% (20 out of 95) unique product features that are not detected by BOW_3. Thus, we investigated which words are detected by ME_1 and not by BOW_3 and found that 70% of the additionally detected product features occur in the dataset fewer than 5 times.

Table 3.7: Confusion Matrix for the Hotel Dataset

	BOW_3	**ME_1**
TP	438	367
TN	8023	8105
FP	140	58
FN	156	277

Thus, a combination of the two approaches is promising because it facilitates the use of the opposing recall and precision and the detection of rarely mentioned product features. In the next section, we apply our proposed ensemble approach, as well as some benchmarks, to these datasets.

3.6.2 Results for EM and benchmarks

In this section, we present the results of our proposed approach as well as those of further benchmarks. Again we applied a 10-fold cross-validation. Tables 3.8 and 3.9 show the results for both datasets.

Table 3.8: Results for the Hotel Dataset

	F1	Precision	Recall
NP	0.0679	0.0680	0.9982
HMM_1	0.2600	0.1521	0.9000
HMM_2	0.3540	0.2206	0.9000
HMM_3	0.2513	0.2353	0.2876
BOW_1	0.6696	0.5271	0.9255
BOW_2	0.7101	0.6087	0.8559
BOW_3	0.7471	0.7563	0.7390
ME_1	0.7303	0.8818	0.6260
ME_2	0.6952	0.8916	0.5741
EM_1	0.7266	0.6053	0.9138
EM_2	0.7768	0.7435	0.8152
EM_3	0.7245	0.6084	0.9012
EM_4	**0.7805**	0.7331	0.8360

Table 3.9: Results for the TV Dataset

	F1	Precision	Recall
NP	0.0721	0.0376	0.9659
HMM_1	0.1599	0.0910	0.8607
HMM_2	0.2345	0.1447	0.8382
HMM_3	0.1742	0.1334	0.2586
BOW_1	0.5960	0.4626	0.8598
BOW_2	0.6394	0.5314	0.8152
BOW_3	0.6178	0.7334	0.5376
ME_1	0.5181	0.9091	0.3679
ME_2	0.5154	0.9154	0.3640
EM_1	0.6466	0.5318	0.8378
EM_2	0.6661	0.7340	0.6131
EM_3	0.6466	0.5318	0.8378
EM_4	**0.6687**	0.7351	0.6169

As expected, the combination of an ME model and a BOW model increases the accuracy substantially. For both datasets, our ensemble approach EM_4 performs best and achieves outstanding performance, surpassing the performance of all the BOW and ME models and the benchmarks. The NP approach has the worst F1 for both datasets and, as expected, the best recall. The recall should be one for both datasets, but a few nouns are tagged as non-nouns, which results in a recall of less than one for both datasets. The hidden Markov chain model based on single words performs twice as well as NP and better than the sequence-based HMM_3. Thus, the structure of a whole sequence is not sufficient to detect product features in customer reviews. HMM_1 predicts many non-noun words, which according to our definition are not features, to be features. Thus, HMM_2, which limits features to be nouns, has the highest accuracy among the HMMs. The BOW models have an F1 that is at least twice as high as that of the HMM models.

3.7 Robustness check

To validate the results and to assess the performance with different training dataset sizes, we applied our proposed approach and the benchmarks to a third dataset. This dataset consists of 200 online customer reviews of different airlines, expressing the users' experience for their flight. Table 3.10 provides an overview of this dataset.

Table 3.10: Overview of the Flight Dataset

	Flight Dataset
Number of reviews	200
Mean review length in words (SD)	126.45 (79.00)
Mean number of sentences (SD)	8.66 (4.55)
Mean cosine similarity between reviews (SD)	0.29 (0.04)
Number of product features	676
Number of unique product features	81
Percentage of unique product features	0.11
Number of reviews without features	37
Mean number of features per review (SD)	3.38 (2.79)
Number of product features with occurrence <5	58
Percentage of product features with occurrence <5	0.71

First, we want to validate the performance in a manner analogous to that used for the hotel and TV dataset. Table 3.11 presents the results of each method based on a 10-fold cross-validation for the flight review dataset. The ensemble approach EM_4 achieves the best accuracy. BOW_3 achieves the highest accuracy among the BOW models. The rare word threshold decreases the performance of the ME models, so ME_1 performs better than does ME_2. Furthermore, HMM_2 is the best-performing HMM, and the naive approach performs the worst. These results are consistent with our previous investigation.

Table 3.11: Results for the Flight Dataset

	F1	Precision	Recall
NP	0.0471	0.0241	0.9925
HMM_1	0.1530	0.0886	0.6275
HMM_2	0.1667	0.0978	0.6275
HMM_3	0.0.800	0.0809	0.0822
BOW_1	0.4892	0.3341	0.9196
BOW_2	0.4920	0.3664	0.7577
BOW_3	0.5464	0.4527	0.6968
ME_1	0.4888	0.7635	0.3683
ME_2	0.4565	0.7728	0.3281
EM_1	0.5060	0.3734	0.7926
EM_2	0.5658	0.4608	0.7400
EM_3	0.5051	0.3732	0.7892
EM_4	**0.5684**	0.4608	0.7491

Next, we investigate the performance for different training dataset sizes. We applied our proposed ensemble approach and the benchmarks to training dataset sizes of 10%, 30%, 50%, 70% and 90% of the complete flight review dataset. A training

dataset size of 10% contains 20 of 200 customer reviews. We applied 10-fold cross-validation for each training dataset size. Figure 3.2 illustrates the performance of each method with different training dataset sizes. The BOW models perform much better than the ME models, especially for small training sets, which indicates that ME models require more customer reviews for training than do BOW models. BOW_1 has the highest accuracy among the BOW models for small training datasets. Small training datasets contain many features that occur fewer than five times due to the limited number of words. Thus, the implementation of a threshold decreases the accuracy of the BOW models when only a few customer reviews are available for training. By contrast, the threshold improves the accuracy as the training dataset size increases because the probability that a word (product feature) is contained in the bag increases. The rare word threshold overcomes this problem. Our proposed ensemble approach always performs best, even for the small training dataset size of only 20 customer reviews. When the training dataset size is at least 30%, EM_4 always performs the best. The performance of the ME models is poor for small training datasets. Thus, the key contributors of the ensemble approaches to the accuracy when the training dataset is small are the BOW models. The hidden Markov chain models benefit from an increasing dataset size if there are few customer reviews, but they still perform much worse than do the BOW models, the ME models and our proposed ensemble models. The baseline approach always results in low accuracy.

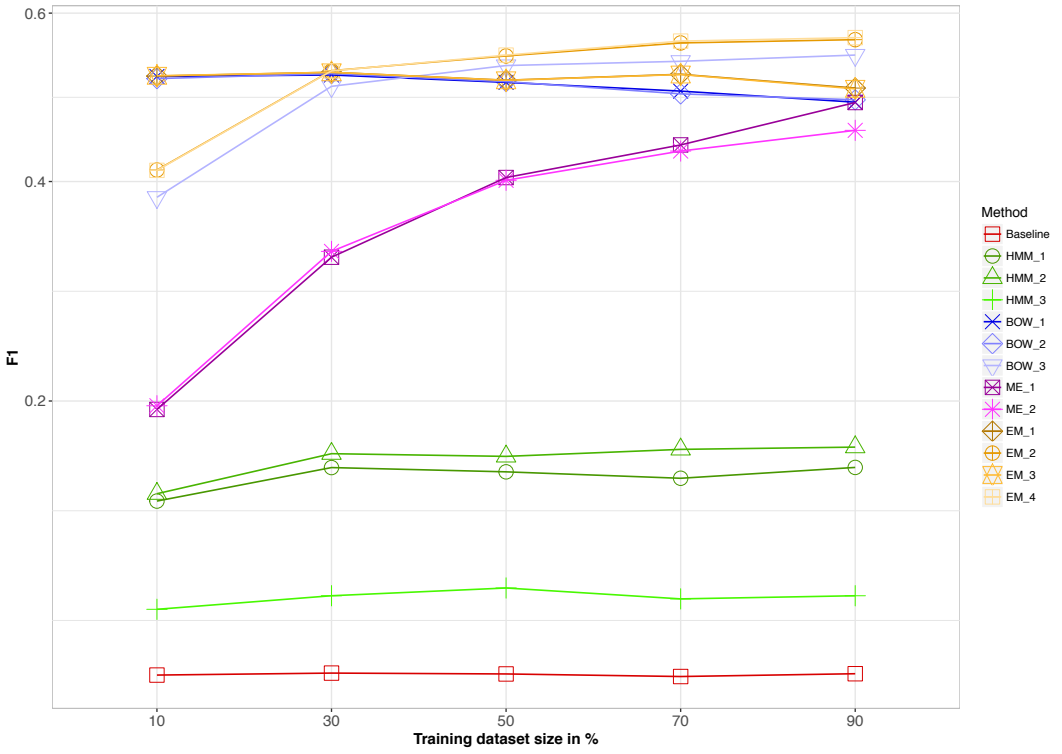

Figure 3.2: F1 for different training dataset sizes

In summary, our assumption that the combination of ME and BOW models will improve the accuracy is confirmed. The combination of a BOW model with a rare word condition and an ME model trained with the context and POS tags achieves outstanding performance for feature detection in online customer reviews. Furthermore, our proposed approach achieves the highest accuracy even when only a few customer reviews are available for training.

3.8 Discussion

In this paper, we propose an ensemble approach based on a BOW model and an ME model. We investigated our approach with three different datasets and product categories and demonstrated that our proposed approach achieves high accuracy, even when the training dataset is small. Our proposed ensemble approach does not use any external dependencies, which are usually not free of charge for commercial use. The proposed approach is easy to implement and easy to customize for other languages by switching the POS tagger. We focused on developing an approach that achieves high accuracy even for a small number of customer reviews because new or small e-commerce platforms have only a few customer reviews per product category. Thus, machine learning techniques, such as deep learning, which require a large training dataset, are difficult to apply. For example, we applied a deep machine learning approach, namely, a feed-forward multilayer artificial neural network, on all three datasets to detect product features, and achieved F1 values of approximately 0.5413 for the hotel dataset, 0.342 for the TV dataset, and 0.3514 for the flight reviews dataset. Thus, the accuracy is much lower than that of the proposed approach, likely due to the small training dataset. Our proposed ensemble approach also achieves high accuracy with a relatively small training dataset. Thus, our proposed approach is especially recommended for e-commerce platforms with few customer reviews. In practical applications, feature detection is a necessary preceding step for several technologies, such as filtering systems improving consumer convenience. Convenience has a strong positive impact on customer purchase decisions (Chen et al., 2010); hence, e-commerce retailers should offer such systems to increase sales. Consumers consider product features when searching for a product that satisfies their demands (Zhu and Zhang, 2010; Brucks et al., 2000). Intuitively, consumers do not know all the product features that determine a product's quality in advance. Thus, consumers would benefit if they are offered a filtering system that suggests important product features for the actual considered product. A proxy for this importance may be the frequency with which a product feature has been discussed in customer reviews because the more important a product feature is, the more customers may have written about it. Thus, retailers should offer a filtering system so that consumers can filter products based on the product overview.

Further, retailers can offer such a filtering system on the product detail page, so consumers do not have to read all the customer reviews but can filter for those that discuss important product features. Thus, consumers can find appropriate products more easily, thereby increasing purchase probability.

Our approach is subject to three major limitations. First, we assume that a feature consists of only one word. This assumption does not support product features consisting of multiple words, such as *remote control* or *room service*. Second, we focus on the extraction of explicit features. OCRs also contain implicit features, which cannot be detected with our approach. Third, the rare word condition may vary for different datasets and must be evaluated before applying our ensemble model.

Finally, two interesting areas for further research exist: first, the extension of our ensemble approach to detect implicit product features; second, the application of our ensemble approach to detect opinion words in OCRs. Thus, an investigation of opinion word detection with an ensemble approach based on a BOW with annotated opinion words and an ME model provides an interesting avenue for further research.

References

Billur Akdeniz, Roger J. Calantone, and Clay M. Voorhees. Effectiveness of Marketing Cues on Consumer Perceptions of Quality: The Moderating Roles of Brand Reputation and Third-Party Information. *Psychology & Marketing*, 30(1):76–89, jan 2013. ISSN 07426046.

Nikolay Archak, Anindya Ghose, and Panagiotis G Ipeirotis. Deriving the Pricing Power of Product Features by Mining Consumer Reviews. *Management Science*, 57(8):1485–1509, 2011. ISSN 0025-1909.

Stephan Bloehdorn and Andreas Hotho. Boosting for Text Classification with Semantic Features. pages 149–166, 2006. ISSN 0302-9743. doi: 10.1007/11899402_10.

Merrie Brucks, Valarie A. Zeithaml, and Gillian Naylor. Price and brand name as indicators of quality dimensions for consumer durables. *Journal of the Academy of Marketing Science*, 28(3):359–374, 2000. ISSN 00920703. doi: 10.1177/0092070300283005.

Samuel W.K. Chan and Mickey W.C. Chong. Sentiment analysis in financial texts. *Decision Support Systems*, 94:53–64, 2017. ISSN 01679236. doi: 10.1016/j.dss.2016.10.006.

Runyu Chen and Wei Xu. The determinants of online customer ratings: a combined domain ontology and topic text analytics approach. *Electronic Commerce Research*, 17(1):31–50, 2017. ISSN 15729362. doi: 10.1007/s10660-016-9243-6.

Ying Hueih Chen, I. Chieh Hsu, and Chia Chen Lin. Website attributes that increase consumer purchase intention: A conjoint analysis. *Journal of Business Research*, 63(9-10):1007–1014, 2010. ISSN 01482963. doi: 10.1016/j.jbusres.2009.01.023.

Verena Dorner, Olga Ivanova, and Michael Scholz. Think Twice Before You Buy! How Recommendations Affect Three-Stage Purchase Decision Processes. 5, 2013.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. Unsupervised named-entity extraction from the Web: An experimental study. *Artificial Intelligence*, 165(1):91–134, jun 2005. ISSN 00043702. doi: 10.1016/j.artint.2005.03.001.

G. Forman. An Extensive Empirical Study of Feature Selection Metrics for Text Classification. *J. Machine Learning Research*, 3:1289–1305, 2003.

Bin Guo and Shasha Zhou. What makes population perception of review helpfulness: an information processing perspective. *Electronic Commerce Research*, 17(4):585–608, 2017. ISSN 15729362. doi: 10.1007/s10660-016-9234-7.

Minqing Hu and Bing Liu. Mining Opinion Features in Customer Reviews. *Proceeding AAAI'04 Proceedings of the 19th national conference on Artifical intelligence*, pages 755–760, 2004a.

Minqing Hu and Bing Liu. Mining and Summarizing Customer Reviews. *Proceeding KDD '04 Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177, 2004b.

Jacob Jacoby, Donald E. Speller, and Carol A. Kohn. Brand Choice Behavior as a Function of Information Load. *Journal of Marketing Research*, 11(1):63–69, feb 1974. ISSN 00222437. doi: 10.2307/3150994.

Niklas Jakob and Iryna Gurevych. Extracting opinion targets in a single-and cross-domain setting with conditional random fields. *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, (October):1035–1045, 2010.

Nan Jing, Tao Jiang, Juan Du, and Vijayan Sugumaran. Personalized recommendation based on customer preference mining and sentiment assessment from a Chinese e-commerce website. *Electronic Commerce Research*, 2017. ISSN 1389-5753. doi: 10.1007/s10660-017-9275-6.

Mangi Kang, Jaelim Ahn, and Kichun Lee. Opinion mining using ensemble text hidden Markov models for text classification. *Expert Systems with Applications*, 0:1–10, 2017. ISSN 09574174. doi: 10.1016/j.eswa.2017.07.019.

Kevin Lane Keller and Richard Staelin. Effects of Quality and Quantity of Information on Decision Effectiveness. *Journal of Consumer Research*, 14:200–213, 1987. doi: 10.2307/2489411.

Aurangzeb Khan, Khairullah Khan, Shakeel Ahmad, Fazal Masood Kundi, Irum Tareen, and Muhammad Zubair Asghar. Lexical Based Semantic Orientation of Online Customer Reviews and Blogs. *Journal of American Science J Am Sci*, 1010(88):143–147, 2014.

Nozomi Kobayashi, Kentaro Inui, and Yuji Matsumoto. Opinion Mining from Web Documents: Extraction and Structurization. *Transactions of the Japanese Society for Artificial Intelligence*, 22(2):227–238, 2007. ISSN 1346-0714. doi: 10.1527/tjsai.22.227.

Naresh K. Malhotra. Reflections on the Information Overload Paradigm in Consumer Decision Making. *Journal of Consumer Research*, 10:436–440, 1984. doi: 10.2307/2488913.

Doaa Mohey. Enhancement Bag-of-Words Model for Solving the Challenges of Sen-

timent Analysis. *International Journal of Advanced Computer Science and Applications*, 7(1):244–252, 2016. ISSN 21565570. doi: 10.14569/IJACSA.2016.070134.

Thomas L. Ngo-Ye and Atish P. Sinha. The influence of reviewer engagement characteristics on online review helpfulness: A text regression model. *Decision Support Systems*, 61(1):47–58, 2014. ISSN 01679236. doi: 10.1016/j.dss.2014.01.011.

Do-Hyung Park and Jumin Lee. eWOM overload and its effect on consumer behavioral intention depending on consumer involvement. *Electronic Commerce Research and Applications*, pages 386–398, 2007. doi: 10.1016/j.elerap.2007.11.004.

Ana-Maria Popescu and Oren Etzioni. Extracting product features and opinions from reviews. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing - HLT '05*, pages 339–346, Morristown, NJ, USA, 2005. Association for Computational Linguistics. doi: 10.3115/1220575.1220618.

Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. Expanding domain sentiment lexicon through double propagation. *IJCAI International Joint Conference on Artificial Intelligence*, pages 1199–1204, 2009. ISSN 10450823.

L. Rabiner and B. Juang. An introduction to hidden Markov models. *IEEE ASSP Magazine*, 3(January):Appendix 3A, 1986. ISSN 0740-7467. doi: 10.1109/MASSP.1986.1165342.

Adwait Ratnaparkhi. Maximum Entropy Models For Natural Language Ambiguity Resolution. *IRCS Technical Reports Series*, 60(March), 1998.

Richard Socher, Alex Perelygin, and Jy Wu. Recursive deep models for semantic compositionality over a sentiment treebank. *Proceedings of the ...*, (October): 1631–1642, 2013. ISSN 1932-6203. doi: 10.1371/journal.pone.0073791.

Gamgarn Somprasertsri and Pattarachai Lalitrojwong. A maximum entropy model for product feature extraction in online customer reviews. *2008 IEEE Conference on Cybernetics and Intelligent Systems*, pages 575–580, 2008. doi: 10.1109/ICCIS.2008.4670882.

Veselin Stoyanov and Claire Cardie. Topic identification for fine-grained opinion analysis. *Proceedings of the 22nd International Conference on Computational Linguistics*, (August):817–824, 2008. doi: 10.3115/1599081.1599184.

Bo Wang and Houfeng Wang. Bootstrapping Both Product Features and Opinion Words from Chinese Customer Reviews with Cross-Inducing. *Proceedings of IJCNLP 2008*, 2008.

Casey Whitelaw, Navendu Garg, and Shlomo Argamon. Using appraisal groups for sentiment analysis. *Proceedings of the 14th ACM international conference*

on Information and knowledge management - CIKM '05, page 625, 2005. ISSN 1947-4040. doi: 10.1145/1099554.1099714.

Bing Xu, Tie-Jun Zhao, De-Quan Zheng, and Shan-Yu Wang. Product features mining based on Conditional Random Fields model. In *2010 International Conference on Machine Learning and Cybernetics*, pages 3353–3357. IEEE, jul 2010. ISBN 978-1-4244-6526-2. doi: 10.1109/ICMLC.2010.5580679.

Boya Yu, Jiaxu Zhou, Yi Zhang, and Yunong Cao. Identifying Restaurant Features via Sentiment Analysis on Yelp Reviews. *arXiv*, pages 1–6, 2017.

Jianxing Yu, Zheng-Jun Zha, Meng Wang, and Tat-Seng Chua. Aspect Ranking : Identifying Important Product Aspects from Online Consumer Reviews. *Computational Linguistics*, pages 1496–1505, 2011. doi: 10.1109/CC.2013.6488828.

Jiaming Zhan, Han Tong Loh, and Ying Liu. Gather customer concerns from online product reviews - A text summarization approach. *Expert Systems with Applications*, 36(2 PART 1):2107–2115, 2009. ISSN 09574174. doi: 10.1016/j.eswa.2007.12.039.

Lei Zhang, Bing Liu, Suk Hwan Lim, and Eamonn O'Brien-Strain. Extracting and Ranking Product Features in Opinion Documents. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 1462–1470, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

Feng Zhu and Xiaoquan (Michael) Zhang. Impact of Online Consumer Reviews on Sales : The Moderating Role of Product and Consumer. *Journal of Marketing*, 74 (2):133–148, 2010. ISSN 0022-2429. doi: 10.1509/jmkg.74.2.133.

Li Zhuang, Feng Jing, and Xiao-Yan Zhu. Movie Review Mining and Summarization. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, CIKM '06, pages 43–50, New York, NY, USA, 2006. ACM. ISBN 1-59593-433-2. doi: 10.1145/1183614.1183625.

4 Filtering based on Customer Reviews: An Investigation of the Impact of Filtering Systems on Purchase Decision Processes

Abstract

Customer reviews provide an important source of information for consumers' purchase decisions. To offer better access to this information, online platforms such as Amazon have begun to implement product- and review filtering systems that are based on terms extracted from customer reviews. In this study, we investigate the effects of product and review filtering systems that are based on customer reviews on consumers' purchase decision processes. We therefore measure the impact of the filtering systems, i.e., product and review filtering, on all three stages of a purchase decision process – screening, evaluation, selection. The results of a laboratory experiment with 114 participants show that i) both systems reduce consumers' effort in making a purchase decision, ii) a product filtering system helps consumers in making a purchase decision that improves their consumer surplus, and iii) a product filtering system allows consumers to be focused on attributes other than quality (e.g., price) when evaluating products. Our findings provide interesting implications for researchers, consumers, and platform providers.

Authors: Tristan Wimmer, Michael Scholz

4.1 Introduction

Several online stores support consumers in their purchase decision processes with decision support systems such as filtering systems or customer review systems (Benlian et al., 2012). Customer review systems are designed to help consumers obtain information about a product's quality prior to purchase. Consumers rely on customer reviews to narrow down their search for products with an acceptable level of quality (Huang et al., 2017; Jang et al., 2012). Filtering systems, in contrast, help consumers to strain information. A combination of both systems seems promising, especially when there are many customer reviews and such reviews contain substantial information required by consumers to evaluate a product's quality. Prior research has

proposed a variety of approaches for automatically extracting product features discussed in customer reviews (Etzioni et al., 2005; Hu and Liu, 2004; Somprasertsri and Lalitrojwong, 2008). These approaches form the technological basis for filtering systems that operate on customer reviews. Although various approaches for product feature extraction have been developed and proposed to be used for filtering customer reviews, investigations of the effects of such filtering systems on consumers' purchase decision processes have fallen short. We propose that the product features extracted from customer reviews can be used for two types of filtering systems – product filtering systems and customer review filtering systems. Product filtering systems help consumers to narrow down a list of available products whereas customer review systems help to evaluate a particular product's quality. Thus, both systems might have different effects on purchase decision processes.

In a controlled laboratory experiment, we investigate the effects of product filtering and customer filtering systems on consumers' purchase decision processes. The results show that i) both systems reduce consumers' effort in making a purchase decision; ii) a product filtering system already affects the purchase decision process in the early stages, whereas a review filtering system only has an effect on the later stages; and iii) the effect of product filtering systems on early purchase decision process stages allows consumers to shift from focusing on evaluating a product's quality to evaluating a product's price in the later stages, which ultimately leads to a purchase of a cheaper but qualitatively equal product.

This study contributes to a vivid stream of research on the effects of software tools on consumers' purchase decision processes. By measuring the effect of product and review filtering systems on all stages of the purchase decision process, this study presents deep insights into the formation of a purchase decision with and without the focal filtering systems. This enables us to, among other things, study the aspects (price and quality) upon which consumers are focused in different stages of the purchase decision process. From a methodical point of view, we present a suggestion of how to measure the evolvement of a purchase decision process based on the screening of available products to making a final purchase decision.

The remainder of the paper is organized as follows. In the next section, we define our objects of investigation – product and review filtering systems. Then, we present the theoretical foundations and the subsequent research model. Thereafter, we describe the laboratory experiment and report the findings. Finally, we discuss our findings with respect to related research and examine the implications for research and practice.

4.2 Product and Review Filtering Systems

Filtering systems are information systems that sort items (filter) from a set of items due to certain criteria defined either by the users of such a system or by the system itself. Within a purchase decision process, the items that are of major interest are product descriptions and customer reviews (Benlian et al., 2012). Product filtering systems allow consumers to either explicitly define constraints (i.e., aspiration levels) such as a maximal price or to implicitly set criteria for filtering out inadequate products. These systems can use different sources of data such as product descriptions, customer reviews and user input (e.g., importance weights or aspiration levels).
Customer reviews aim at helping consumers in diagnosing the quality of a particular product. However, they are threatened by their own success due to the high number of available customer reviews, and hence it is difficult for consumers to find relevant customer reviews (Scholz and Dorner, 2013). For example, Amazon.com, introduced a review filtering system with which consumers can easily filter reviews by using the frequently used words as filter criteria.
In this study, we consider product filtering systems and review filtering systems that use customer reviews as a database. More specifically, the product features discussed in customer reviews can be used for filtering products or reviews. The product features that are available for a particular product and the level of quality are important to consumers to know to judge the utility a product offers to her. Thus, filtering products based on product features extracted from customer reviews might help a consumer to find prospective products more easily, whereas filtering reviews based on product features facilitates the diagnosing of the quality of a particular product. For both filtering systems, the data used for filtering as well as the objects that will be filtered are shown in Figure 4.1.

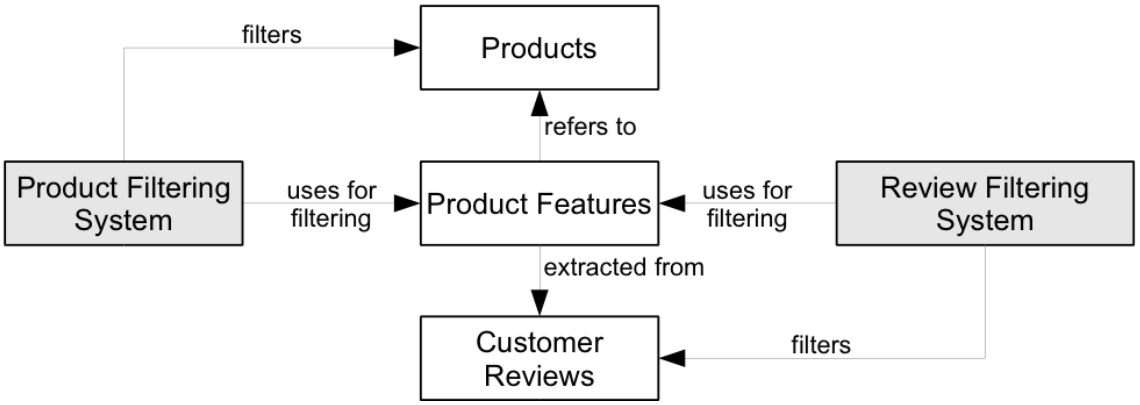

Figure 4.1: Product Filtering and Review Filtering System

Both filtering systems likely affect consumers' purchase decision processes but in different ways. Product filtering systems affect the purchase decision process at a

rather early stage whereas review filtering systems might have an effect on the later stages.

4.3 Theoretical Foundation

In this section, an overview is provided of the theoretical foundations required to answer our research questions. First, we explain the purchase decision process and describe each stage of this process in detail. Second, we present the related work and show how specific filtering systems, such as recommender systems, influence each stage of the purchase decision process.

4.3.1 Purchase Decision Process

Consumers use various decision rules to search and evaluate available products and finally make a purchase decision (Hauser, 2014). In addition, the criteria used by consumers to stop the search and evaluation process are divergent. Some consumers aim at finding the product with the highest utility whereas others are satisfied with a product that surpasses an individually defined utility threshold (Chowdhury et al., 2009). However, there is consensus in the marketing and information systems literature that consumers' purchase decision making in general is a process that consists of multiple stages (Dorner et al., 2013; Hauser and Wernerfelt, 1990; Wu and Rangaswamy, 2003). Several researchers propose a two-stage purchase decision process (Gilbride and Allenby, 2004; Hauser and Wernerfelt, 1990; Roberts and Lattin, 1991) whereas others assume a three-stage decision making process (Dorner et al., 2013; Wu and Rangaswamy, 2003).

We follow Wu and Rangaswamy (2003) and assume a three-stage purchase decision process that starts with the screening of available products (awareness set). All products that pass the screening stage form the consideration set (Hauser and Wernerfelt, 1990; Shocker et al., 1991). Consumers use rather simple, noncompensatory decision rules in the screening stage to eliminate products from the awareness set that are unattractive (Hauser, 2014; Hauser et al., 2010; Moe, 2006).

The products in the consideration set are evaluated in detail in a second stage. The result of this evaluation stage is the choice set consisting of all products a consumer is willing to purchase (Wu and Rangaswamy, 2003). Consumers typically use other and more complex decision rules in their evaluation than in the screening stage (Moe, 2006). Thus, evaluating a particular product is on average more time-consuming than screening a particular product.

Finally, the consumer selects a product from the choice set. The selection criterion in this stage does not need to be the overall utility. Some consumers accept the risk of selecting a product that does not provide the highest utility to reduce the cognitive

effort for evaluating products and making a final choice (Chowdhury et al., 2009; Johanson and Payne, 1985; Payne et al., 1992). Figure 4.2 illustrates the purchase decision process.

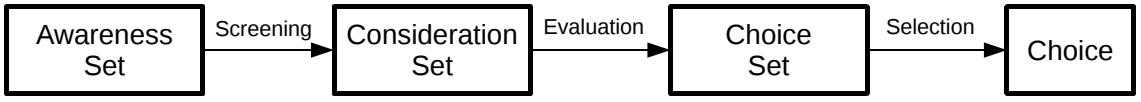

Figure 4.2: Purchase Decision Process

Several personal and contextual factors have been found to affect the purchase decision process (Chakravarti and Janiszewski, 2003; Häubl and Trifts, 2000; Suh, 2009). We discuss the influence of filtering systems as a particular contextual factor on the size, quality, and price of the result of each of the three process stages in the next section.

4.3.2 Influence of Filtering Systems on Purchase Decision Processes

In particular, prior work investigated the influence of recommender systems on purchase decision processes (Dellaert and Häubl, 2012; Parra and Ruiz, 2009; Pathak et al., 2010). Recommender systems can be considered to be product filtering systems that filter products with mainly historical or implicit consumer input (Ghoshal et al., 2015).

These systems help consumers in forming their consideration and choice sets faster and more accurately (Dorner et al., 2013; Häubl and Trifts, 2000). The average utility of a product filtered with a recommender system is higher than the average utility of a product from the awareness set. However, the marginal expected utility of screening the next recommended product decreases with the number of previously inspected products in the presence of a recommender system (Dellaert and Häubl, 2012). Screening the next product is hence less meaningful when using a recommender system than without using such a system. As a consequence, consumers' consideration sets are smaller and the products in the consideration set are more homogeneous with respect to their utility (Dorner et al., 2013; Häubl and Trifts, 2000; Parra and Ruiz, 2009). Recommender systems shift consumers' focus from screening products toward evaluating products in detail (Dellaert and Häubl, 2012; Lenton and Francesconi, 2010). There is furthermore evidence that recommender systems positively influence a consumer's propensity to convert to a purchaser (Benlian et al., 2012; Pathak et al., 2010).

There is only sparse evidence about the impact of other filtering systems on purchase

decision processes. Dabrowski and Acton (2013) demonstrate that a relaxation of user-specific filter constraints increases the quality of the finally selected product. However, they did not investigate how the decision process is affected by a product filtering system compared to a situation with no product filtering system.

4.4 Research Model

Product filtering systems are available just at the beginning of the purchase decision process and hence might affect all three stages of the process. Review filtering systems are available only if the consumer has selected a product to consider in detail. Thus, we expect that these filtering systems will have an effect only on the formation of the choice set as well as on the final choice.

4.4.1 Effects of Using a Product Filtering System

4.4.1.1 Consideration Set

The possibility of filtering out some (inappropriate) products reduces the size of the awareness set and increases the average utility of the products within the awareness set. The number of products that have a chance to be in the consideration set is thus reduced if a product filtering system is available. Furthermore, consumers have some prior information about the utility of the products' being in the awareness set: they know that the products respect the filter criteria. The marginal benefits of including the next product in the consideration set are lower than in a situation without a product filtering system because of a higher homogeneity of filtered products (Häubl and Trifts, 2000). We thus expect that consumers who use a product filtering system will have smaller consideration sets than consumers who do not use such a system.

H1a: The use of a product filtering system leads to a reduction in the number of products in the consideration set.

We furthermore expect that the average quality of the products in the consideration set is higher if consumers can use a product filtering system because products that do not meet the desired criteria can be filtered out.

H1b: The use of a product filtering system leads to a higher average quality of the products in the consideration set.

If the available filters are not correlated with the price of the products, we do not expect that the average price of the products in the consideration set will be affected by the availability of a product filtering system.

H1c: The use of a product filtering system does not lead to different prices of products in the consideration set if the filter criteria are not correlated with the price of the products.

4.4.1.2 Choice Set

Consumers seriously consider the products in the consideration set during the evaluation stage to determine their utilities and to decide which products to transfer to their choice set. The size of the choice set is therefore limited by the size of the consideration set. Because we expect smaller consideration sets when consumers use a product filtering system, we can also expect smaller choice sets if a product filtering system is used. However, we also expect that the products in the consideration set to be of higher average quality when consumers use a product filtering system. Furthermore, the products in the consideration set are more homogeneous with respect to their quality when inappropriate products can be filtered out. Smaller consideration sets typically also indicate lower heterogeneity in the considered products' utility (Häubl and Trifts, 2000). Prior work has shown that the difficulty in selecting products increases when the heterogeneity of the products in the consideration set decreases (Dhar, 1997; Dhar and Simonson, 2003). Dorner et al. (2013) demonstrate that the probability that a product is transferred from the consideration set to the choice set increases when consumers can use decision support systems that decrease the heterogeneity of the products' utility in the consideration set. We thus expect that consumers who use a product filtering system will have a larger choice set relative to their consideration set size.

H2a: The use of a product filtering system leads to an increase in the choice set size relative to the consideration set size.

Prior research cites evidence that the average quality of "goal-satisfying" products (i.e., products in the choice set) is lower when search costs are low, due to the availability of decision support systems (Diehl, 2005; Diehl et al., 2003). Product filtering systems reduce search costs because they filter out inadequate products. Nevertheless, we expect a higher quality of the products included in the consideration set and hence the products that can be in the choice set (see H1b). Furthermore, filtering ensures that the minimum average quality is above the average quality when no filtering is available. Filtering signals that the products in the awareness set have a higher average quality so that consumers might put less effort into evaluating the quality of the products in the consideration set. We thus expect no difference in terms of the average quality between choice sets from consumers who use a product filtering system and consumers who do not.

H2b: The use of a product filtering system does not affect the average quality of the products in the choice set.

If the quality of the consideration set is already high due to the possibility of filtering out inadequate products, consumers can put more concentration into the price or price-quality evaluation than on the evaluation of pure quality. Diehl (2005) provide

evidence that consumers concentrate more on price than on quality if the relative importance of price is higher than the slope of the quality on price. We therefore expect that the use of a product filtering system will reduce the average price of the alternatives in the choice set.

H2c: The use of a product filtering system reduces the price of the products in the choice set.

4.4.1.3 Choice

We argue that a product filtering system does not affect the quality of the products included in the choice set. Thus, selecting a product from a choice set formed with the help of a product filtering system will lead to a quality that is comparable to selecting from a choice set formed without access to a product filtering system.

H3a: The use of a product filtering system does not affect the quality of the finally selected product.

The lower price of the products in the choice set will consequently lead to a lower price of the selected product when using a product filtering system. We hence expect a higher consumer surplus when consumers have the opportunity to use a product filtering system.

H3b: The use of a product filtering system reduces the price of the selected product.

4.4.1.4 Time of the Purchase Decision Process

We expect that consumers' effort in screening products and building a consideration set is significantly reduced due to i) a smaller awareness set and ii) a higher utility of the products in the result set. We further expect that this lower consumer effort will reduce the total time a consumer invests in the purchase decision process.

H4: The use of a product filtering system reduces the total time of the purchase decision process.

4.4.2 Effects of Using a Review Filtering System

4.4.2.1 Consideration Set

A review filtering system supports consumers in filtering customer reviews for a particular product. Customer reviews are considered to evaluate the quality of a product. We hence expect that review filtering systems will not influence the size, quality or price of the products in the consideration set.

H5a: The use of a review filtering system does not affect the number of products in the consideration set.

H5b: The use of a review filtering system does not affect the average quality of the products in the consideration set.

H5c: The use of a review filtering system does not affect the average price of the products in the consideration set.

4.4.2.2 Choice Set

A review filtering system makes it easier for consumers to find customer reviews that discuss the product features that are important to them. This will likely reduce the time necessary to determine whether a product is worth buying. We expect that the presence of a review filtering system will induce a shift in the purchase decision process from evaluating only a few products in depth towards evaluating more products with less effort. Dellaert and Häubl (2012) found support for such a shift in the inverse direction in the presence of a recommendation system that assists consumers especially in the screening stage. Because a review filtering system supports consumers in the evaluation stage, we propose that consumers can evaluate more products with the same effort than they would without the presence of a review filtering system.

H6a: The use of a review filtering system increases the number of products in the choice set.

Choice sets likely contain products of similar quality (Lehmann and Pan, 1994) because consumers can better justify a compromise than an extreme product (Simonson, 1989). A review filtering system provides the possibility of easily identifying customer reviews that discuss the quality features that are important for a consumer in a particular situation. We thus expect that consumers will likely choose products with a higher quality in the choice set.

H6b: The use of a review filtering system increases the average quality of the products in the choice set.

Price is a typical search attribute and can be evaluated without any uncertainty based on retailer or manufacturer information. Customer reviews are not necessary and do not help in the evaluation of the price of a particular product. Therefore, we expect no effect on the price of the products in the choice set in the presence of a review filtering system.

H6c: The use of a review filtering system does not affect the average price of the products in the consideration set.

4.4.2.3 Choice

We argue that the quality of the products included in the choice set is positively affected by a review filtering system (H6b). We hence expect that the presence of a

review filtering system will improve the quality of the finally chosen product.

H7a: The use of a review filtering system will increase the quality of the selected product.

Because customer reviews typically do not help to evaluate the price of a product, a review filtering system does neither affect the price of the products in the choice set nor the price of the finally selected product.

H7b: The use of a review filtering system will not affect the price of the selected product.

4.4.2.4 Time of the purchase decision process

In the presence of a review filtering system, consumers need to spend less effort in the evaluation of a product. Consequently, we expect that consumers evaluate more products (see hypothesis H6a). Decision rules to evaluate a product are more complex than the rules applied to screen a product (Gensch, 1987; Moe, 2006). Thus, we expect that a consumer must invest much more effort in evaluating a product than she must invest in screening a product. Therefore, although we expect that consumers will build larger choice sets when they have access to a review filtering system, we also expect that they will need less time to make a final selection.

H8: The use of a review filtering system reduces the total time of the purchase decision process.

Figure 4.3 summarizes our research model.

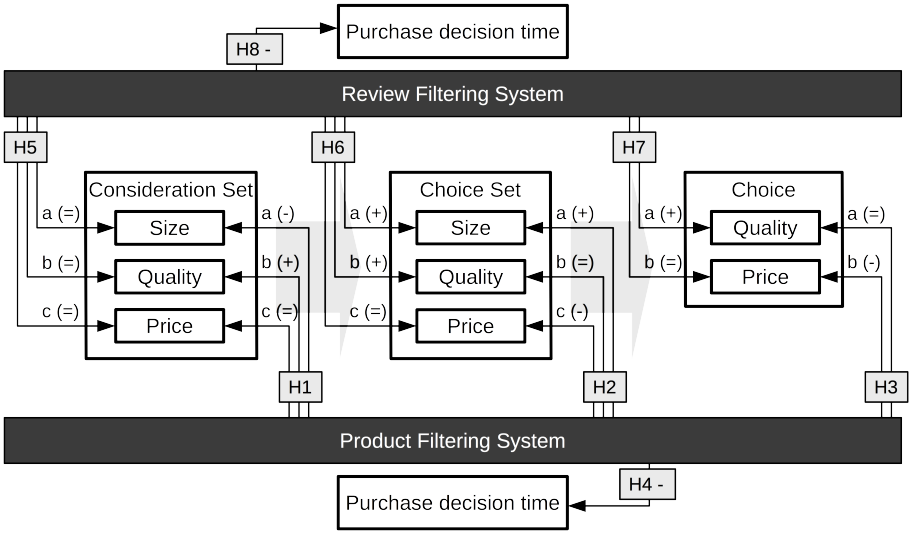

Figure 4.3: Research Model

4.5 Empirical Investigation

We conducted a laboratory experiment in the PAULA[1] lab at the University of Passau to investigate our research model. In this section, we describe the treatments, the experimental procedure and the sample used for the experiment.

4.5.1 Treatments

We investigated the effects of product filtering and review filtering systems on consumers' purchase decision processes with the 2 x 2 between-subjects design shown in Table 4.1. The basis of all treatments is a self-implemented hotel booking platform that operates on a database including 640 hotels located in Vienna. We chose hotels for two main reasons. First, hotels are a very popular good, with which we could assume all our participants to be reasonably familiar. Second, evaluating a hotel description is possible without any specific expert knowledge. The hotels are in a price range between 43 and 712 Euros with a median of 103 Euros per night. Each hotel is described by a price, customer reviews, and an average customer rating. Hotel descriptions were extracted from a large real-existing booking platform, and the names as well as locations were anonymized.

Table 4.1: Experimental Design

Treatment	Product Filtering System	Review Filtering System
1	–	–
2	X	–
3	–	X
4	X	X

Our self-implemented hotel booking platform supports all three stages of the purchase decision process. A list of all available hotels (awareness set) is presented on the start page. The treatments that allow product filtering (i.e., treatments 2 and 4) present a list of product features frequently discussed in the customer reviews of all 640 hotels (see Figure 4.4). By clicking on a particular product feature, the list of hotels is filtered such that only hotels with customer reviews discussing the selected feature are shown. Hotels are described by an identifier, the average customer rating and the price in the hotel list.
Participants can click on each hotel to obtain a description that consists of the hotel identifier, the price, average customer ratings with respect to seven categories (comfort, equipment, location, free Wi-Fi, staff, price performance ratio and overall), and a list of all customer reviews for the focal hotel (see Figure 4.5). Treatments that

[1] The PAULA lab is a professionally equipped and managed laboratory with computer cubicles that allowed us to control several potential confounding variables such as communication between the participants.

allow review filtering (i.e., treatments 3 and 4) present the same list of frequently discussed product features as presented on the hotel overview page by treatments 2 and 4. Participants can filter the customer reviews by clicking on a specific product feature.

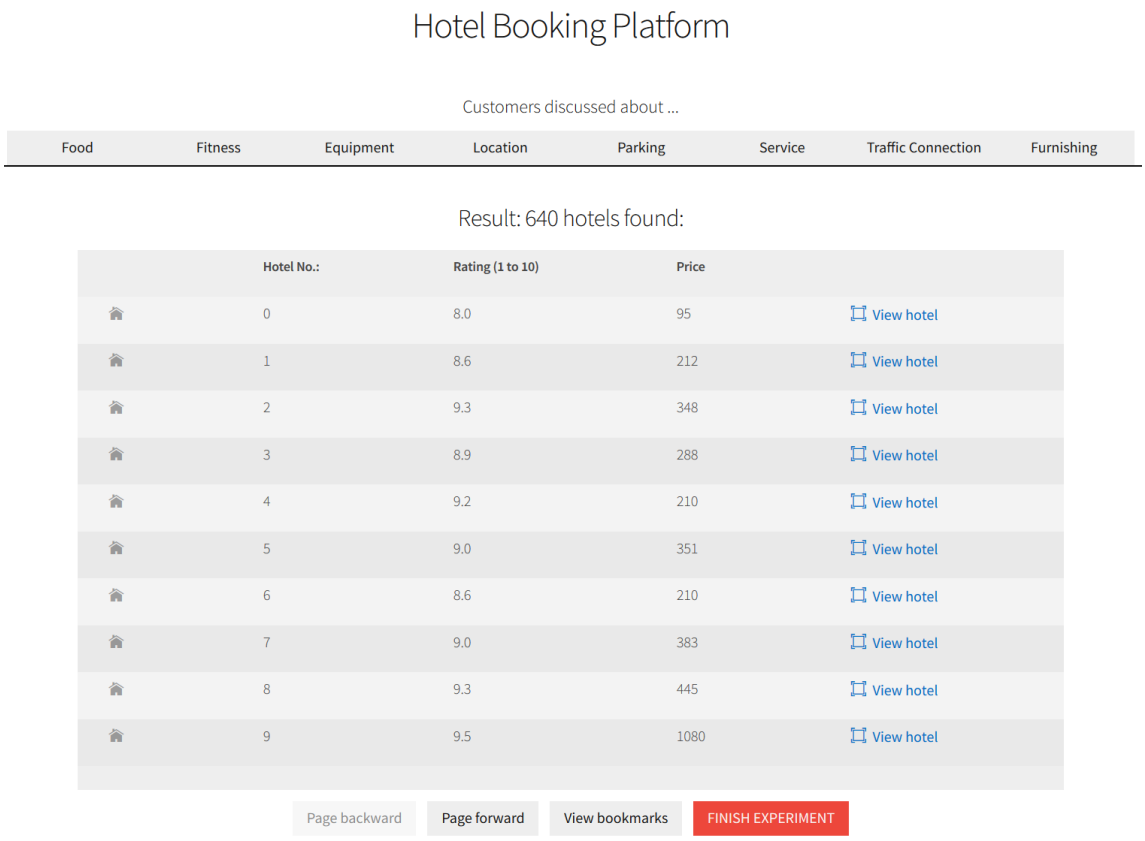

Figure 4.4: Translated Mock-up of the Hotel Overview Page with Product Filtering Possibility

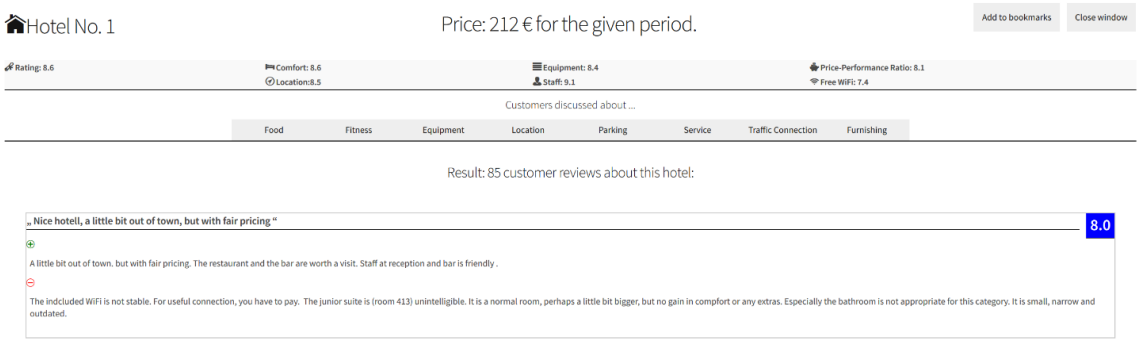

Figure 4.5: Translated Mock-up of the Hotel Detail Page with Review Filtering Possibility

Each hotel can be bookmarked. A link to the bookmark page is available on each other page so that hotels can be removed from the bookmark list at any time.

The product features were manually extracted from all customer reviews in our database. Because many extracted product features (e.g., parking space, parking lot) referred to one and the same feature (e.g., parking), we aggregated the extracted product features and finally came up with a list of eight product features (food, fitness, equipment, location, parking, service, traffic connection, and furnishing).

4.5.2 Experimental Procedure

The experiment proceeded as follows. Each participant was randomly assigned to one of the four treatments. Each participant received an introduction sheet including i) a description of the study context, ii) instructions for the experiment, and iii) a short user's manual for the self-implemented booking platform. The participants were instructed to assume that they are planning to travel to Vienna for two nights. They should look for a hotel offering parking and fitness possibilities and costs not more than 250 Euro in total. There was no time limit for the hotel searching task, and there was no need to come up with a final choice (a no-choice option was allowed). After reading the instructions, the participants could open the laptops and begin searching for an adequate hotel with one of the described treatments.
The participants could finish the hotel search at any time by clicking on a button labeled "Finish Experiment," which was available on the hotel overview page. When clicking on this button, the participants were instructed to have all hotels in the bookmark list they have positively evaluated and to be basically willing to book. After the hotel search task, the participants answered an additional questionnaire with questions about their finally selected hotel, their experience with hotel booking platforms, the number of holiday trips per year, and demographic characteristics.

4.5.3 Variables

According to our research model, we measured different variables for each step of the purchase decision process. More specifically, we measured the size of the consideration and choice set, the quality of the alternatives in consideration and choice set and of the final choice and the price of all alternatives in the consideration and choice set as well as the price of the finally selected hotel. We therefore define the consideration set as consisting of all hotels for which the detail page has been requested. The choice set is defined by all hotels that are finally on the bookmark list. Because a hotel can only be added to the bookmark list from the detail page, we can ensure that all hotels that are in the choice set are also in the consideration set. The final choice is a hotel from the choice set (i.e., bookmark list) that the participant would definitely book for a two-day trip.
We simply counted the number of hotels in the consideration and choice sets to obtain the size of the two sets. The quality was measured by counting the number of

hotels offering parking and fitness possibilities. This number is divided by two times the consideration or choice set size to obtain a value scaled in 0; 1] that expresses how many hotels fulfill the requirements. As the price for the consideration (choice) set, we use the average price across all hotels in the consideration (choice) set.

4.5.4 Pretest

We carried out one-on-one pretests with four students who did not take part in the final experiment. Afterwards, the participants were asked to write down their opinions of and thoughts on every step of the experiment. The resulting comments were respected in the final experiment. We also tested the manipulation checks and found that all pretest participants correctly applied the filtering systems.

4.5.5 Sample

We invited 140 undergraduate and graduate students of a public university in Germany to take part in a laboratory experiment. A total of 114 participated in and successfully completed the experiment. A total of 29 participants used the first treatment (no filtering), 25 used the second treatment (product filtering), 29 used the third treatment (review filtering) and 31 used the fourth treatment (product and review filtering). Differences in treatment group sizes are due to the random assignment of participants to treatments. Each participant was paid 10 Euros. The participants' age, gender, booking platform usage and number of holiday trips is presented in Table 4.2 for each treatment. ANOVA tests indicate no significant differences for all variables.

Table 4.2: Sample Characteristics and ANOVA Results (p-value) for Differences between Treatment Groups

Variable	Treatment 1	Treatment 2	Treatment 3	Treatment 4	p-value
Age	22.32	22.10	21.84	22.86	0.571
Females	67.74%	58.62%	72.00%	75.86%	0.504
Booking platform usage	86.67%	82.76%	72.00%	93.33%	0.186
Holiday trips per year	2.68	3.10	3.40	2.97	0.539
N	31	29	25	29	–

4.6 Data Analysis and Results

4.6.1 Manipulation Check

Each participant visited the detail page of at least three hotels. Therefore, we can assume that each participant had the chance to use the product filter and the review filter if available. We also analyzed how often our participants used a product filter criterion or review filter criterion. The results are presented in Table 4.3.

Table 4.3: Usage of Filter Systems

Treatment	Product Filter Criteria			Review Filter Criteria		
	Min	Median	Max	Min	Median	Max
Treatment 1 – No Filter Systems	0	0	0	0	0	0
Treatment 2 – Product Filter System	2	5	17	0	0	0
Treatment 3 – Review Filter System	0	0	0	12	135	238
Treatment 4 – Both Filter Systems	1	4.5	16	4	32	256

All participants who had the opportunity to use the product filtering and/or the review filtering system filtered products and/or reviews by defining at least one criterion. Thus, we can assume that our participants correctly perceived the presence or absence of the filtering systems.

4.6.2 Descriptive Analysis

Table 4 provides an overview of our measured purchase decision process variables. The participants who used treatment 1 (no filter system) finally chose the most expensive hotel across all treatments. They also spent more time until they finally found an adequate hotel. The finally selected hotel of the participants in treatment group 4 (both filter systems are available) is on average approximately 24 Euros cheaper than the hotel selected by the participants in treatment group 1.

Table 4.4: Mean (Standard Deviation) of our Purchase Decision Process Variables

Variable		Treatment 1	Treatment 2	Treatment 3	Treatment 4
Consideration Set	Size	67.16 (69.62)	12.79 (6.06)	90.96 (50.97)	24.27 (33.32)
	Quality	0.53 (0.08)	0.96 (0.06)	0.52 (0.14)	0.92 (0.13)
	Price	200.87 (31.70)	195.12 (16.91)	200.86 (20.82)	188.01 (14.71)
Choice Set	Size	4.71 (2.69)	6.10 (3.29)	9.68 (6.14)	10.87 (0.26)
	Quality	0.93 (0.13)	0.99 (0.03)	0.99 (0.04)	0.99 (0.03)
	Price	194.01 (18.69)	179.53 (12.33)	190.95 (12.93)	180.14 (10.99)
Choice	Quality	0.97 (0.12)	1.00 (0.00)	0.98 (0.10)	1.00 (0.00)
	Price	202.29 (33.19)	180.38 (21.05)	191.48 (26.65)	178.23 (18.47)
Total Time (in minutes)		50.48 (10.63)	37.48 (13.52)	37.74 (13.90)	33.33 (11.79)

The consideration set size and the choice set size are relatively high compared to the set sizes reported for other products (Dorner et al., 2013; Häubl and Trifts, 2000). One reason might be that the hotels were sparsely described, thus our participants did not need to read and digest much product information.

4.6.3 Effects on the Consideration Set

The consideration set size is an over-dispersed count variable. We hence conducted a negative binomial regression to investigate the effects of our treatments on the consideration set sizes of the participants in the experiment. We defined the quality

to be a binomial variable and thus analyzed the differences with respect to the quality with a logistic regression. We used a linear regression with an OLS estimator for comparing the price of the hotels in the consideration set across our four treatment groups. The results of all regression analyses for investigating the effects on the consideration set are presented in Table 4.5.

Table 4.5: Effects on the Consideration Set

Variable	Size	Quality	Price
Intercept	4.207 (0.135)***	-0.721 (0.027)***	200.865 (3.992)***
Product Filtering	-1.658 (0.200)***	0.671 (0.591)***	-5.748 (5.742)
Review Filtering	0.303 (0.202)	-0.012 (0.376)	-0.003 (5.974)
Product Filtering x Review Filtering	0.337 (0.286)	-0.152 (0.076)	-7.100 (8.318)

Significance codes: $***$ p < 0.001, $**$ < 0.01, $*$ < 0.05

We expected a reduced consideration set size for the participants who used the product filtering system. Thus, we can support H1a. Two reasons for the reduction of the consideration set in the presence of a product filtering system are possible. First, filtering reduces the awareness set such that several products might not be available to be included in the consideration set after filtering. Second, the filtered products will have a higher utility reducing the effort to find a product that surpasses a specific utility threshold. We conducted another analysis to investigate the reason for the reduction in the consideration set size. Therefore, we created a variable that expresses the consideration set size depending on the number of filtered products. More specifically, we computed the fraction of the number of considered hotels and the actual number of filtered products. Because the participants were able to change the filter criteria, we computed this fraction for each filter step and used the mean overall fractions as the adjusted consideration set size. For example, if a participant activates the filter criterion "location" and then screens three products, the fraction is three divided by the number of filtered products – in this case, 322. If she then deactivates the filter criterion and screens two other hotels, we calculate the second fraction as $\frac{2}{640}$. The adjusted consideration set size finally is $\left(\frac{3}{322} + \frac{2}{640}\right)/2 = 0.00622$. We compared the four treatment groups with respect to the adjusted consideration set size with a logistic regression and did not find a significant effect of the filtering systems (p > 0.5). This indicates that the reduction of the awareness set instead of the higher utility is responsible for the smaller consideration sets of the participants who used the product filtering system.

Product filtering helps to filter out products with a low quality (i.e., products that do not fulfill some defined constraints). Thus, the quality of the products in the consideration set has been found to be higher for participants who used the product filtering system. Hence, we can support H1b.

The product filtering system used in the empirical investigation offers consumers

the possibility of using product features frequently discussed in customer reviews as filter criteria. Because the price is a search attribute, it is typically not discussed in customer reviews. Hence, the product filter criteria are rather not correlated with price, and we did not expect any effect of the product filtering system on the price of the products in the consideration set. The results in Table 4.5 support this hypothesis (H1c).
The reviews for a hotel can be filtered just after the hotel successfully passed the screening stage and has been included in the consideration set. We thus did not expect any effects of the review filtering system on the consideration set. Table 4.5 indicates that the review filtering system indeed did not affect the size, quality or price of the consideration set, supporting hypotheses H5a, H5b, and H5c.

4.6.4 Effects on the Choice Set

We also found the choice set size to be over-dispersed and hence analyzed the effect of the four treatments on the choice set size with a negative binomial regression. We used the consideration set size as an additional covariate in this regression because we assume the filtering systems have an effect on the choice set size relative to the consideration set size. We applied a logistic regression for investigating the effects on the quality and a linear regression to identify the effects on the price of the products in the choice set. The results of these regression analyses are presented in Table 4.6.

Table 4.6: Effects on the Choice Set

Variable	Size	Quality	Price
Intercept	1.029 (0.141)***	-0.075 (0.084)	194.006 (2.543)***
Product Filtering	0.683 (0.171)***	0.070 (0.113)	-14.478 (3.658)***
Review Filtering	0.531 (0.154)***	0.064 (0.106)	-3.062 (3.806)
Product Filtering x Review Filtering	-0.107 (0.209)	-0.064 (0.141)	3.678 (5.300)
Consideration Set Size	0.007 (0.001)***	–	–

Significance codes: $***$ p < 0.001, $**$ < 0.01, $*$ < 0.05

Both filtering systems positively influence the choice set size. Consumers consider more products in detail relative to the number of products in the consideration set. Hypotheses H2a and H6a are thus supported. As expected, we also did not find any effect of the product filtering system on the quality of the products in the choice set (H2b). The review filtering system should help consumers to more easily evaluate a product's quality in detail. This is underscored by the fact that the consumers who used the review filtering system evaluated more products relative to their consideration set size. However, the average quality is not significantly higher when using a review filtering system than not. This might be because the high quality of the products in the choice set of the participants who did not use a review filtering system (see Table 4.4). Even the participants who used neither the product

filtering nor the review filtering system had choice sets with an average quality of 93% . With 99%, the average quality is higher for the participants who used the review filtering system; however, this rather small difference is not statistically significant. We hence found no support for H6b.

The price of the products in the choice set is on average approximately 14 Euros lower in the presence of a product filtering system. Because we did not find a significant effect of product filtering on the price of the products in the consideration set, we can assume that consumers who used the product filtering system have switched from a quality focus to a price focus. We estimated the importance weights used by the consumers for screening and evaluating with a random-effects model to obtain further insights into the focus-shift from quality to price in the evaluation stage by the usage of a product filtering system. We use the decision of whether a product from the awareness set (consideration set) is included in the consideration set (choice set) as the dependent variable. We further use the normalized price and quality as the covariates. The normalization ensures that price and quality are both in the interval [0;1] and that the highest price (lowest quality) level is 0 whereas the lowest price (highest quality) level is 1. This enables us to directly compare the regression coefficients for price and quality. We further estimate participant-specific random intercepts because each participant screened multiple products and evaluated multiple products. The additional fixed-effect analyses are depicted in Table 4.7 and Table 4.8.

Table 4.7: Importance Weights for Screening Products

Variable	Treatment 1	Treatment 2	Treatment 3	Treatment 4
Intercept	-12.344 (0.521)***	-24.295 (1.507)***	-10.412 (0.483)***	-24.294 (1.170)***
Price	9.975 (0.522)***	15.434 (1.519)***	8.425 (0.464)***	17.548 (1.157)***
Quality	1.954 (0.081)***	8.893 (0.364)***	1.791 (0.075)***	7.218 (0.221)***
Price- Quality- Ratio	5.105	1.736	4.704	2.431

Significance codes: $***$ p < 0.001, $**$ < 0.01, $*$ < 0.05

Table 4.8: Importance Weights for Evaluating Products

Variable	Treatment 1	Treatment 2	Treatment 3	Treatment 4
Intercept	-11.943 (2.984)***	-29.822 (4.842)***	-27.821 (3.455)***	-31.124 (3.891)***
Price	5.175 (3.144)	24.587 (4.566)***	17.589 (3.456)***	22.302 (3.766)***
Quality	6.389 (0.507)***	7.256 (1.617)***	11.749 (0.945)***	11.484 (1.172)***
Price- Quality- Ratio	0.810	3.389	1.497	1.942

Significance codes: $***$ p < 0.001, $**$ < 0.01, $*$ < 0.05

The results show that the products' price is relatively more important in the evaluation than in the screening stage in the presence of a product filtering system (treatment 2). Without any filtering system, our participants were mainly focused on price in the screening stage (i.e., when building the consideration set), and they considered quality to be slightly more important than price in the evaluation stage

(i.e., when building the choice set). With the possibility of using a product filtering system, the importance of a product's price was over three times higher than its quality in the evaluation stage. Thus, a product filtering system causes a shift from quality-focused product evaluations to price-focused product evaluations. Thus, we can support hypothesis H2c.

In support of H6c, we did not find any effect of the review filtering system on the price of the hotels in the choice set.

4.6.5 Effects on Choice

All participants finally found a hotel they were willing to book for a two-day stay in Vienna. We analyzed the effects of the filtering systems on the quality of the finally selected product with a logistic regression and the effects on the price of the chosen product with a linear regression.

Table 4.9: Effects on Choice

Variable	Quality	Price
Intercept	3.401 (1.016)***	202.29 (4.59)***
Product Filtering	18.16 (5428.33)	-21.91 (6.60)**
Review Filtering	0.49 (1.75)	-10.81 (6.87)
Product Filtering x Review Filtering	-0.49 (7612.57)	8.66 (9.56)

Significance codes: $***$ p < 0.001, $**$ < 0.01, $*$ < 0.05

As expected, product filtering has no significant influence on the quality of the finally chosen product (see Table 4.9). Hypothesis H3a is thus supported. Almost all participants chose a product that fulfilled all necessary constraints (i.e., parking and fitness possibility). We hence did not find a significant difference between the product quality of the participants who used the review filtering system and that of the participants who did not. Hypothesis H7a is not supported.
The participants who had the possibility of filtering hotels finally selected a hotel that is on average more than 20 Euros cheaper than the hotels selected by the participants who could not filter hotels. This difference is statistically significant and supports hypothesis H3b. We also found support for hypothesis H7b because the review filtering system did not significantly affect the price of the finally chosen hotel.

4.6.6 Effects on Total Time

We analyzed the effect of the filtering systems on the total time the participants required to come up with a final choice with a linear regression. The results are depicted in Table 4.10.

Table 4.10: Effects on Total Time

Variable	Total Time
Intercept	50.48 (2.234)***
Product Filtering	-13.00 (3.21)***
Review Filtering	-12.74 (3.34)***
Product Filtering x Review Filtering	8.58 (4.66)

Significance codes: $***$ p < 0.001, $**$ < 0.01, $*$ < 0.05

Both filtering systems support consumers in their purchase decision process and reduce the effort that must be invested by the consumers to decide. In support of hypotheses H4 and H8, the participants who used a product filtering and/or a review filtering system needed significantly less time for the complete purchase decision process.

4.6.7 Summary

A summary of all hypotheses tests is presented in Table 5.11. Our results show that product filtering and review filtering systems that are based on product features extracted from customer reviews do not improve the quality of the finally selected product. However, both systems drastically reduce the time needed by consumers to invest to make a final decision.

Table 4.11: Hypotheses Tests

System		Hypothesis	Result
Product Filtering	Consideration Set	**H1a:** The use of a product filtering system leads to a reduction in the number of products in the consideration set.	supported
		H1b: The use of a product filtering system leads to a higher average quality of the products in the consideration set.	supported
		H1c: The use of a product filtering system does not lead to different prices of products in the consideration set if the filter criteria are not correlated with the price of the products.	supported

	Choice Set	**H2a:** The use of a product filtering system leads to an increase in the choice set size relative to the consideration set size.	supported
		H2b: The use of a product filtering system does not affect the average quality of the products in the choice set.	supported
		H2c: The use of a product filtering system reduces the price of the products in the choice set.	supported
	Choice	**H3a:** The use of a product filtering system does not affect the quality of the finally selected product.	supported
		H3b: The use of a product filtering system reduces the price of the selected product.	supported
	Time	**H4:** The use of a product filtering system reduces the total time of the purchase decision process	supported
Review Filtering	Consideration Set	**H5a:** The use of a review filtering system does not affect the number of products in the consideration set.	supported
		H5b: The use of a review filtering system does not affect the average quality of the products in the consideration set.	supported
		H5c: The use of a review filtering system does not affect the average price of the products in the consideration set.	supported

	Choice Set	**H6a:** The use of a review filtering system increases the number of products in the choice set.	supported
		H6b: The use of a review filtering system increases the average quality of the products in the choice set.	not supported
		H6c: The use of a review filtering system does not affect the average price of the products in the consideration set.	supported
	Choice	**H7a:** The use of a review filtering system will increase the quality of the selected product.	not supported
		H7b: The use of a review filtering system will not affect the price of the selected product.	supported
	Time	**H8:** The use of a review filtering system reduces the total time of the purchase decision process.	supported

Furthermore, a product filtering system ensures that the products considered in early stages of the purchase decision process are already of high quality. This allows consumers to be more focused on price than quality when evaluating a product, which ultimately leads to a selection of products that are significantly cheaper. On average, the participants who used the product filtering finally selected a hotel that is approximately 9% cheaper than those selected by participants who did not use a filtering system. Figure 4.6 depicts the quality trend, and Figure 4.7 presents the price trend over the purchase decision process for product filtering, review filtering and no filtering.

The consumers increase their consideration set quality by using a product filtering system. The filtering of products furthermore reduces the number of products that can possibly be added to the consideration set, which results in smaller consideration sets of higher quality. Therefore, a product filtering system facilitates consumers in focusing on other attributes (e.g., price or nonfilterable attributes) in the evaluation stage.

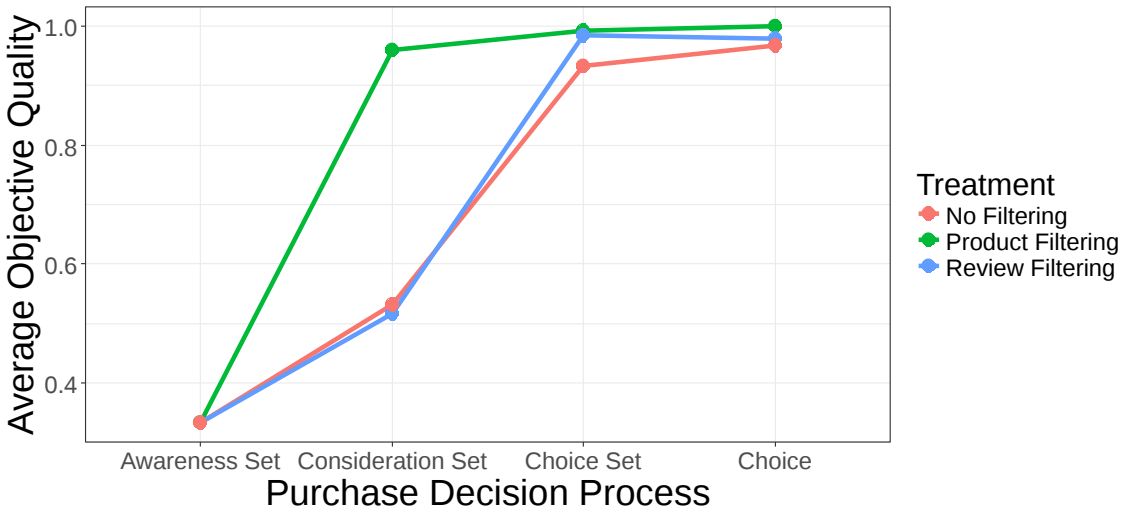

Figure 4.6: Quality Trend for Filtering Systems

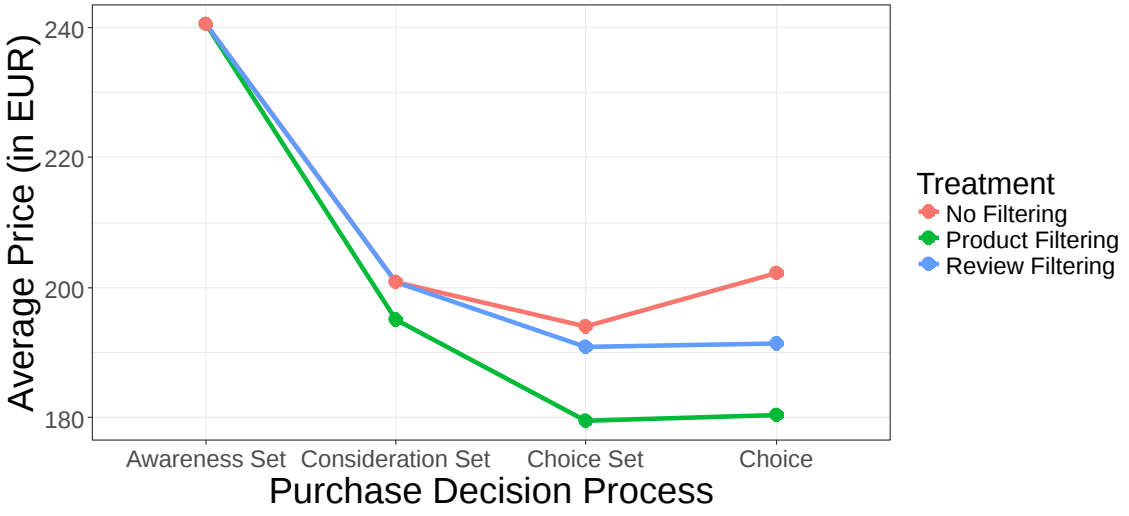

Figure 4.7: Price Trend for Filtering Systems

4.7 Discussion

This study investigates the effects of product filtering and review filtering systems operating on customer reviews on consumers' purchase decision processes. More specifically, we investigated the impact of the two filtering systems on consumers' consideration set, choice set and final choice in a controlled laboratory experiment. Whereas product filtering systems affect the purchase decision process in the first stage (screening), review filtering systems have been found to affect only the second (evaluation) stage. Although the two filtering systems have not been found to affect the quality of the finally selected product, the stages before a consumer can make a final purchase decision are very different. Both systems allow the consumers to make

a final purchase decision in maximally 75% of the time required by the consumers without a filtering system. The consumers who use a product filtering system especially save time by forming significantly smaller consideration sets, whereas the consumers who use a review filtering system save time with a simplified product evaluation. The higher quality of products in the consideration by the consumers who use a product filtering system furthermore causes a shift in the importance weights for price and quality.

4.7.1 Research Implications

Our findings have several important implications for understanding consumers' purchase decision-making behavior. First, in line with Dellaert and Häubl (2012), we argue that consumers evaluate more products if decision support systems are available that either make the evaluation easier or that make the preparation of a set of products easier to evaluate. The decision about whether a next product should be evaluated in detail depends on the effort a consumer must invest and the expected utility of the product to be evaluated next (Dellaert and Häubl, 2012; Hauser and Wernerfelt, 1990). A review filtering system makes it easier for consumers to diagnose the quality of a certain product and hence reduces the effort a consumer must invest. A product filtering system, in contrast, ensures a high utility of products because low-utility products can be filtered out. Second, we found evidence for a shift in the aspects mainly considered in the evaluation stage. Without any filtering system, consumers' consideration sets consist of a high proportion of low quality products. Thus, consumers place high importance on evaluating a product's quality when forming the choice set. Because quality in the consideration set is much higher on average when consumers use a product filtering system, they consider quality only with rather low importance in the evaluation stage and can thus focus on other aspects such as price.
Finally, we argue in line with Wu and Rangaswamy (2003) and Dorner et al. (2013) that purchase decision processes consist of three steps (screening, evaluation, and selection), and we present an approach for measuring the results of each of the three steps (i.e., consideration set, choice set, and choice). Our approach uses only behavioral data and is applicable for many online platforms. We define the consideration set to include products that were selected from a product overview page that provides product details. As such, a selection typically corresponds to a click; the measurement of the consideration set is rather easy. The choice set is defined to consist of all products that were seriously considered for purchase. We used the concept of a bookmark list to measure the choice set. A choice set is comparable to a list of bookmarks in that they are used for objects with some importance like products in a choice set. Adding a product to the bookmark list also corresponds to a click

that can be easily observed in an experiment. A consumer's choice is defined as the product that is purchased or marked for purchase (e.g., added to a shopping cart) and can also be measured by analyzing a consumer's clicks in an online platform.

4.7.2 Managerial Implications

Our findings also provide implications for consumers and online platform providers. Consumers benefit from product- and review filtering systems in the reduced effort they need to invest to make a purchase decision. The reduced effort will likely improve consumers' satisfaction with an online platform, which will ultimately lead to an increase in consumers' loyalty (Cyr, 2008). Consumers who use a product filtering system furthermore gain a higher consumer surplus. The participants who used the product filtering system in our experiment had a consumer surplus of approximately 70 Euros, whereas the participants who did not use any filtering system had only a consumer surplus of approximately 48 Euros. This higher consumer surplus is likely to additionally improve consumers' loyalty to an online platform.
Thus, we recommend that online platform providers should implement product filtering- and review filtering systems to reduce consumers' efforts as much as possible and simultaneously improve consumers' surplus. Although review filtering systems do not significantly affect the price of the finally chosen product, such systems reduce the time that must be invested by a consumer to make a purchase decision. Online platform providers should furthermore offer the possibility for consumers to store products on a bookmark list. This possibility allows online platform providers to obtain deeper insights into the purchase decision processes of their customers. For example, they can analyze the attribute levels that determine why one product has been added to the bookmark list and another one not. Such an analysis will help platform providers to derive marketing strategies and make personalized offers.

4.7.3 Limitations and Future Research

This paper has some limitations that warrant future research. First, we performed a laboratory experiment to control for confounding variables and improve internal validity. Consumers' purchase decision process is subject to several confounding variables such as the influence of additional product information, which may have an impact on the measured effects of product- and review filtering systems on real online platforms. Future research should hence investigate the effects of product and review filtering systems in field experiments to improve the generalizability of our findings. Second, we investigated the effects of the filtering systems only with one product category. The presented effects might differ in their strength for different product categories. Investigating the effects of product filtering based on customer reviews and review filtering systems on purchase decision processes for different

product categories is thus an interesting avenue for future research. Third, although this study offers important insights into the evolvement of a purchase decision in the presence and absence of a product filtering and a review filtering system, no insights are possible about how the addressed filtering systems lead to cognitive and emotional changes of the purchase decision-makers. Biodata, such as fMRI scans or EEG data, are required to obtain deeper insights into the cognitive and emotional state of consumers during a purchase decision process (vom Brocke and Liang, 2014). We encourage researchers to conduct neuroscience studies and map the data of these studies with our study.

References

Alexander Benlian, Ryad Titah, and Thomas Hess. Differential Effects of Provider Recommendations and Consumer Reviews in E-Commerce Transactions: An Experimental Study. *Journal of Management Information Systems*, 29(1):237–272, 2012. ISSN 0742-1222.

Amitav Chakravarti and Chris Janiszewski. The Influence of Macro-Level Motives on Consideration Set Composition in Novel Purchase Situations. *Journal of Consumer Research*, 30(2):244–258, 2003. ISSN 0093-5301. doi: 10.1086/376803. URL http://jcr.oxfordjournals.org/cgi/doi/10.1086/376803.

Tilottama G. Chowdhury, S. Ratneshwar, and Praggyan Mohanty. The time-harried shopper: Exploring the differences between maximizers and satisficers. *Marketing Letters*, 20(2):155–167, 2009. ISSN 09230645. doi: 10.1007/s11002-008-9063-0.

Dianne Cyr. Modeling Web Site Design Across Cultures: Relationships to Trust, Satisfaction, and E-Loyalty. *Journal of Management Information Systems*, 24(4): 47–72, 2008. ISSN 0742-1222. doi: 10.2753/MIS0742-1222240402. URL http://www.tandfonline.com/doi/full/10.2753/MIS0742-1222240402.

MacIej Dabrowski and Thomas Acton. The performance of recommender systems in online shopping: A user-centric study. *Expert Systems with Applications*, 40 (14):5551–5562, 2013. ISSN 09574174. doi: 10.1016/j.eswa.2013.04.022. URL http://dx.doi.org/10.1016/j.eswa.2013.04.022.

Benedict G.C Dellaert and Gerald Häubl. Searching in Choice Mode: Consumer Decision Processes in Product Search with Recommendations. *Journal of Marketing Research*, 49(2):277–288, 2012. ISSN 0022-2437. doi: 10.1509/jmr.09.0481. URL http://journals.ama.org/doi/abs/10.1509/jmr.09.0481.

Ravi Dhar. Consumer Preference for a No-Choice Option. *Journal of Consumer Research*, 24(2):215–231, 1997. ISSN 0093-5301. doi: 10.1086/209506. URL https://academic.oup.com/jcr/article-lookup/doi/10.1086/209506.

Ravi Dhar and Itamar Simonson. The Effect Choice on Choice of Forced. *Journal of Marketing Research*, 40(2):146–160, 2003. ISSN 0022-2437. URL http://journals.ama.org/doi/abs/10.1509/jmkr.40.2.146.19229.

Kristin Diehl. When Two Rights Make a Wrong: Searching Too Much in Ordered Environments. *Journal of Marketing Research*, 42(3):313–322, 2005. ISSN 0022-2437. doi: 10.1509/jmkr.2005.42.3.313. URL http://journals.ama.org/doi/abs/10.1509/jmkr.2005.42.3.313.

Kristin Diehl, Laura J. Kornish, and John G. Lynch. Smart Agents: When Lower Search Costs for Quality Information Increase Price Sensitivity. *Journal of Consumer Research*, 30(June):56–71, 2003. ISSN 1556-5068. doi: 10.2139/ssrn.340040. URL http://www.ssrn.com/abstract=340040.

Verena Dorner, Olga Ivanova, and Michael Scholz. Think Twice Before You Buy! How Recommendations Affect Three-Stage Purchase Decision Processes. *Thirty Fourth International Conference on Information Systems*, 5, 2013.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. Unsupervised named-entity extraction from the Web: An experimental study. *Artificial Intelligence*, 165(1):91–134, jun 2005. ISSN 00043702. doi: 10.1016/j.artint.2005.03.001.

Dennis H. Gensch. A Two-Stage Disaggregate attribute choice model. *Marketing Science*, 6(3):223–239, 1987.

Abhijeet Ghoshal, Subodha Kumar, and Vijay Mookerjee. Impact of recommender system on competition between personalizing and non-personalizing firms. *Journal of Management Information Systems*, 31(4):243–277, 2015. ISSN 1557928X. doi: 10.1080/07421222.2014.1001276.

Timothy J. Gilbride and Greg M. Allenby. A Choice Model with Conjunctive, Disjunctive, and Compensatory Screening Rules. *Marketing Science*, 23(3): 391–406, 2004. ISSN 0732-2399. doi: 10.1287/mksc.1030.0032. URL http://pubsonline.informs.org/doi/10.1287/mksc.1030.0032.

Gerald Häubl and Valerie Trifts. Consumer Decision Making in Online Shopping Environments : The Effects of Interactive Decision Aids Consumer Decision Making in Online Shopping Environments : The Effects of Interactive Decision Aids. *Marketing Science*, 19(1):4–21, 2000. ISSN 0732-2399. doi: 10.1287/mksc.19.1.4.15178.

John R. Hauser. Consideration-set heuristics. *Journal of Business Research*, 67 (8):1688–1699, 2014. ISSN 01482963. doi: 10.1016/j.jbusres.2014.02.015. URL http://dx.doi.org/10.1016/j.jbusres.2014.02.015.

John R. Hauser and Birger Wernerfelt. An Evaluation Cost Model of Consideration Sets. *Journal of Consumer Research*, 16(4):393–408, 1990. ISSN 0093-5301. doi: 10.1086/209225. URL https://academic.oup.com/jcr/article-lookup/doi/10.1086/209225.

John R Hauser, Olivier Toubia, Theodoros Evgeniou, Rene Befurt, and Daria Dzyabura. Disjunctions of Conjunctions, Cognitive Simplicity, and Consideration Sets. *Journal of Marketing Research*, 47(3):485–496, 2010. ISSN 0022-2437. doi: 10.1509/jmkr.47.3.485. URL http://journals.ama.org/doi/abs/10.1509/jmkr.47.3.485.

Minqing Hu and Bing Liu. Mining Opinion Features in Customer Reviews. *Proceeding AAAI'04 Proceedings of the 19th national conference on Artifical intelligence*, pages 755–760, 2004.

Jianxiong Huang, Wai Fong Boh, and Kim Huat Goh. A Temporal Study of the Effects of Online Opinions: Information Sources Matter. *Journal of Management Information Systems*, 34(4):1169–1202, 2017. ISSN 1557928X. doi: 10.1080/07421222.2017.1394079.

Sungha Jang, Ashutosh Prasad, and Brian T. Ratchford. How consumers use product reviews in the purchase decision process. *Marketing Letters*, 23(3):825–838, 2012. ISSN 09230645. doi: 10.1007/s11002-012-9191-4.

Eric J. Johanson and John W. Payne. Effort and Accuracy in Choice. *Management Science*, 31(4):395–414, 1985.

Donald R. Lehmann and Yigang Pan. Context Effects, New Brand Entry, and Consideration Sets. *Journal of Marketing Research*, 31(3):364–374, 1994. ISSN 00222437. doi: 10.2307/3152223. URL http://www.jstor.org/stable/3152223?origin=crossref.

Alison P. Lenton and Marco Francesconi. How humans cognitively manage an abundance of mate options. *Psychological Science*, 21(4):528–533, 2010. ISSN 09567976. doi: 10.1177/0956797610364958.

Wendy W Moe. An Empirical Two-Stage Choice Model with Varying Decision Rules Applied to Internet Clickstream Data. *Journal of Marketing Research*, 43 (4):680–692, 2006. ISSN 0022-2437. doi: 10.1509/jmkr.43.4.680. URL http://journals.ama.org/doi/abs/10.1509/jmkr.43.4.680.

José F. Parra and Salvador Ruiz. Consideration sets in online shopping environments: the effects of search tool and information load. *Electronic Commerce Research and Applications*, 8(5):252–262, 2009. ISSN 15674223. doi: 10.1016/j.elerap.2009.04.005. URL http://dx.doi.org/10.1016/j.elerap.2009.04.005.

Bhavik Pathak, Robert Garfinkel, Ram D. Gopal, Rajkumar Venkatesan, and Fang Yin. Empirical Analysis of the Impact of Recommender Systems on Sales. *Journal of Management Information Systems*, 27(2):159–188, 2010. ISSN 0742-1222. doi: 10.2753/MIS0742-1222270205. URL http://www.tandfonline.com/doi/full/10.2753/MIS0742-1222270205.

John W. Payne, James R. Bettman, and Eric J. Johanson. Behavioral decision research: A Constructive processing perspective. *Annual Review of Psychology*, 43(1):87–131, 1992.

John H. Roberts and James M. Lattin. Development and Testing of a Model of Consideration Set Composition. *Journal of Marketing Research*, 28(4):429, 1991. ISSN 00222437. doi: 10.2307/3172783. URL http://www.jstor.org/stable/3172783?origin=crossref.

Michael Scholz and Verena Dorner. The recipe for the perfect review?: An investigation into the determinants of review helpfulness. *Business and Information Systems Engineering*, 5(3):141–151, 2013. ISSN 18670202.

Allan D. Shocker, Moshe Ben-Akiva, Bruno Boccara, and Prakash Nedungadi. Consideration set influences on consumer decision-making and choice: Issues, models, and suggestions. *Marketing Letters*, 2(3):181–197, 1991. ISSN 1573059X. doi: 10.1007/BF02404071.

Itamar Simonson. Choice Based on Reasons: The Case of Attraction and Compromise Effects. *Journal of Consumer Research*, 16(2):158–174, 1989. ISSN 0093-5301. doi: 10.1086/209205. URL https://academic.oup.com/jcr/article-lookup/doi/10.1086/209205.

Gamgarn Somprasertsri and Pattarachai Lalitrojwong. A maximum entropy model for product feature extraction in online customer reviews. *2008 IEEE Conference on Cybernetics and Intelligent Systems*, pages 575–580, 2008. doi: 10.1109/ICCIS.2008.4670882.

Jung-Chae Suh. The Role of Consideration Sets in Brand Choice: The Moderating Role of Product Characteristics. *Psychology & Marketing*, 26(6):534–550, 2009.

Jan vom Brocke and Ting-Peng Liang. Guidelines for Neuroscience Studies in Information Systems Research. *Journal of Management Information Systems*, 30 (4):211–234, 2014. ISSN 0742-1222. doi: 10.2753/MIS0742-1222300408. URL http://www.tandfonline.com/doi/full/10.2753/MIS0742-1222300408.

Jianan Wu and Arvind Rangaswamy. A Fuzzy Set Model of Search and Consideration with an Application to an Online Market. *Marketing Science*, 22 (3):411–434, 2003. ISSN 07322399. doi: 10.1287/mksc.22.3.411.17738. URL http://mktsci.journal.informs.org/content/22/3/411.abstract.

5 Online Product Descriptions – Boost for your Sales

Abstract

Product descriptions are a source of information online consumers can use to reduce product uncertainty. Recent research provides evidence that consumers favor using information from other consumers, such as customer reviews, over information provided by the retailer or manufacturer, such as product descriptions. We complement this research and show that the presence of product descriptions significantly influences products' sales and that this influence decreases with an increasing number of customer reviews. We furthermore demonstrate that a product description's information amount positively affects a product's sales. The number of customer reviews available for a product also moderates the effect of the information amount of a product description on sales.

Authors: Tristan Wimmer, Michael Scholz

5.1 Introduction

Online consumers face a barrier in physical experience of products. While consumers in offline markets can touch the product of their choice, online consumers hardly can evaluate a product's physical characteristics prior to purchase. Consumers in online as well as offline markets typically perceive uncertainty in purchase decision processes (Akerlof, 1970; Dimoka et al., 2012; Overby and Jap, 2009). Pavlou et al. (2007) investigated reasons for perceived product uncertainty by considering the relationship between sellers and consumers as a principal-agent problem. Amongst others, they identified information asymmetry as one of the most important determinants of perceived uncertainty. This information asymmetry refers to the seller or the product and finally leads to seller uncertainty or product uncertainty (Pavlou et al., 2007; Ghose, 2009). Seller uncertainty is defined as consumer's difficulty to predict a seller's behavior in the future whereas product uncertainty refers to the difficulty to evaluate a product's quality prior to purchase (Dimoka et al., 2012; Hong and Pavlou, 2014). Consumers in offline markets can inspect a product's physical characteristics and can get personal advice from the seller. Consumers in online stores, such as Amazon or Staples, predominantly have two sources of information to learn about a product's characteristics: customer reviews and product descriptions

(Ghose and Han, 2014). Consumers typically have a higher trust in information from other consumers than from a producer or a marketplace (Benlian et al., 2012). Recent research thus has intensively investigated the effect of customer reviews, as one source of information, on consumers' purchase decision processes and retailers' sales (Forman et al., 2008; Purnawirawan, 2014; Zhu and Zhang, 2010). These studies demonstrate that customer reviews affect purchase decisions and ultimately sales to a large extent. The more reviews are available, the lower is a consumer's perceived product uncertainty (Cui et al., 2012; Ehrmann and Schmale, 2008). Consumers tend to consult other information sources (e.g., product descriptions, third-party assurances) if no or only a few customer reviews are available. Dimoka et al. (2012) found evidence that product descriptions reduce product uncertainty most significantly among several information sources, such as product descriptions, product inspections, history reports and product warranties. Further, Detlor et al. (2003) found evidence, that product descriptions are important in pre-purchase online information seeking. Hence, product descriptions are another important source for reducing product uncertainty with two major advantages compared to customer reviews: First, product descriptions are also available in the absence of customer reviews, because they are provided by the seller. Thus, product descriptions are often the only source of information consumers' can digest to learn about a product's characteristics prior to purchase. And second, sellers can control product descriptions and hence the extent to which they reduce product uncertainty. Despite these major advantages, product descriptions and their impact on sales has been sparsely examined in recent research. We thus focus on the effect of product descriptions on the reduction of product uncertainty. We follow Pavlou et al. (2007) and assume that a lower product uncertainty leads to higher sales and analyze the impact of product descriptions on sales. Our empirical investigation shows that products with a product description generate higher sales than products without a description. We provide evidence that the information amount of product descriptions is positively correlated with products' sales. With this paper, we contribute to recent research by (1) examining the effect of the presence and the information amount of product descriptions on sales, (2) distinguishing between the effect of product descriptions written by a retailer and product descriptions written by a manufacturer, (3) investigating the interaction effect of product descriptions and customer reviews on sales.

The remainder of this paper is organized as follows. In the next section, we discuss the theoretical background on product uncertainty and its reduction. Afterwards, we describe our research model followed by a description of our empirical evaluation. We then present the result of the empirical evaluation and of some robustness checks. We conclude this paper with a summary of our results and a discussion of the

implications for researchers and practitioners.

5.2 Theoretical Background

5.2.1 Product Uncertainty

Product uncertainty is defined as consumers' difficulty to evaluate a product's characteristics prior to purchase (Luo et al., 2012). The higher the variance of product characteristics the higher is the perceived product uncertainty (Hong and Pavlou, 2014). A consumer who wants to purchase a new electric toothbrush might feel uncertain about whether a particular toothbrush is controllable via a mobile application. If there is no toothbrush with mobile application support available, there is no variance for this characteristic. Thus, in the case that none of the toothbrushes provides mobile application support there is no product uncertainty. As soon as there are toothbrushes with and without mobile application support available, consumers perceive product uncertainty to a particular extent about the quality of that mobile application support.

Product uncertainty has negative implications for both, sellers and consumers. Recent research has demonstrated that the higher the product uncertainty, the lower is the price premium that can be charged for a particular product (Dimoka et al., 2012). Product uncertainty furthermore negatively affects sales (Pavlou et al., 2007) and the number of product returns (Hong and Pavlou, 2014). Consumers' transaction costs increase with the perceived product uncertainty because consumers need to invest search costs in order to reduce product uncertainty (Liang and Huang, 1998). Furthermore, a rising product uncertainty decreases consumers' purchase intention significantly (Pavlou et al., 2007). Sellers and consumers have a keen interest in reducing product uncertainty. Therefore, sellers and consumers provide information in form of product descriptions or customer reviews to reduce product uncertainty in online stores.

5.2.2 Reduction of Product Uncertainty in Online Stores

Online consumers want to learn about a product's characteristics before purchase in order to reduce product uncertainty. In offline stores, consumers can reduce product uncertainty by inspecting a product itself and requesting individual advice from the seller. Inspecting products prior to purchase is typically not possible in online stores. Thus, online consumers gather information about a product's characteristics from different information sources, such as product descriptions and customer reviews (Akdeniz et al., 2013; Dorner et al., 2013). Customer reviews are peer-generated product evaluations that typically consist of a product rating and an optional textual description of the experiences with the product (Mudambi and Schuff, 2010;

Scholz and Dorner, 2013). Recent research has shown that the existence of customer reviews reduces product uncertainty and thereby improves consumers' purchase probability (Dorner et al., 2013). Customer reviews might vary significantly in the product characteristics they discuss. The probability that a consumer can learn about some certain characteristic increases with the number of provided customer reviews (Liu, 2006). Mudambi and Schuff (2010) furthermore find evidence that the length of the textual description of a customer review is positively correlated with the perceived helpfulness of the review. Longer reviews presumably discuss more product characteristics. Archak et al. (2011) show that reviews discussing more product characteristics are more influential on consumers' purchase decisions. Similarly, Scholz and Dorner (2013) find support that reviews with a higher information amount are more helpful for consumers. Reviews with a higher helpfulness are more likely to reduce product uncertainty (Mudambi and Schuff, 2010). Helpfulness has been furthermore shown to positively affect a retailer's sales (Forman et al., 2008). In summary, product reviews are an important source of information. They influence consumers' decision-making processes by reducing information asymmetries and thus by reducing product uncertainty. In contrast to customer reviews, the effect of product descriptions on reducing consumers' product uncertainty has been sparsely analyzed in existing research. Dimoka et al. (2012) investigate the influence of product descriptions on product uncertainty in online car auctions. They provide evidence that the influence of product descriptions is nearly twice as much as that of third-party assurances, such as (car) inspections, history reports or product warranties. Ghose and Han (2014) find evidence that the length of product descriptions is positively correlated with sales.

In summary, existing research demonstrates that customer reviews and product descriptions contribute to reduce product uncertainty. An investigation of the effect of the existence of product descriptions as well as the interaction of product descriptions and customer reviews on sales is missing so far.

5.3 Research Model

Recent research has illustrated that product uncertainty significantly influences sales (Pavlou et al., 2007). Factors positively influencing product uncertainty are hence likely to negatively influence sales. In the following, we will use sales as proxy for product uncertainty because product sales are easily observable. Product descriptions have been found to be one source for reducing product uncertainty (Dimoka et al., 2012). If no or not enough information is available for reducing product uncertainty, consumers refrain to buy, especially high-priced products (Kim and Krishnan, 2015). The availability of product descriptions helps consumers to learn

about product characteristics and thereby reduce product uncertainty. According to Pavlou et al. (2007), this will finally lead to higher sales.

H1: Products with a product description generate on average more sales than products without a product description.

The information amount transported in a product description significantly varies across the products. Descriptions discussing more product characteristics are more likely to help consumers reducing product uncertainty. Recent research has provided evidence that customer reviews, as another source of information for consumers, are perceived as more helpful if they discuss more product characteristics that are not widely discussed in other customer reviews (Scholz and Dorner, 2013; Otterbacher, 2008). Such an effect has been shown for app descriptions (Ghose and Han, 2014). We expect a similar effect also for product descriptions and thus hypothesize that a product's sales are increasing in the product description's information amount.

H2: The higher the information amount of a product's description, the higher are its sales on average.

Recent research provides ample evidence that the number of customer reviews available for a particular product positively influences this product's sales (Ehrmann and Schmale, 2008; Chen et al., 2004). Consumers prefer customer reviews over product descriptions because the retailer or manufacturer generates the latter (Benlian et al., 2012). Product descriptions communicate a positive picture about a product whereas customer reviews also point to a product's drawbacks. However, product descriptions exist, in contrast to customer reviews, even if no consumer has bought or evaluated the product. The more reviews are available, the higher is the probability that a particular consumer will find enough information to reduce product uncertainty and the higher will be the probability to purchase a product. Recent research has demonstrated that the number of reviews positively affects sales (Forman et al., 2008; Duan et al., 2008). We therefore propose a decreasing impact of a product's description on sales when there is an increasing number of customer reviews available.

H3: The influence of the availability of a product description on a product's sales is moderated by the number of reviews. The more reviews a product has, the lower is the effect of the availability of a description on sales.

Similarly, the impact of a description's information amount might be also moderated by the number of reviews available for a product.

H4: The influence of a description's information amount on sales is moderated by the number of reviews. The more reviews a product has, the lower is the effect of its descriptions' information amount on sales.

Our hypotheses are summarized in Figure 5.1. We describe the empirical evaluation with which we test the derived hypotheses in the following section.

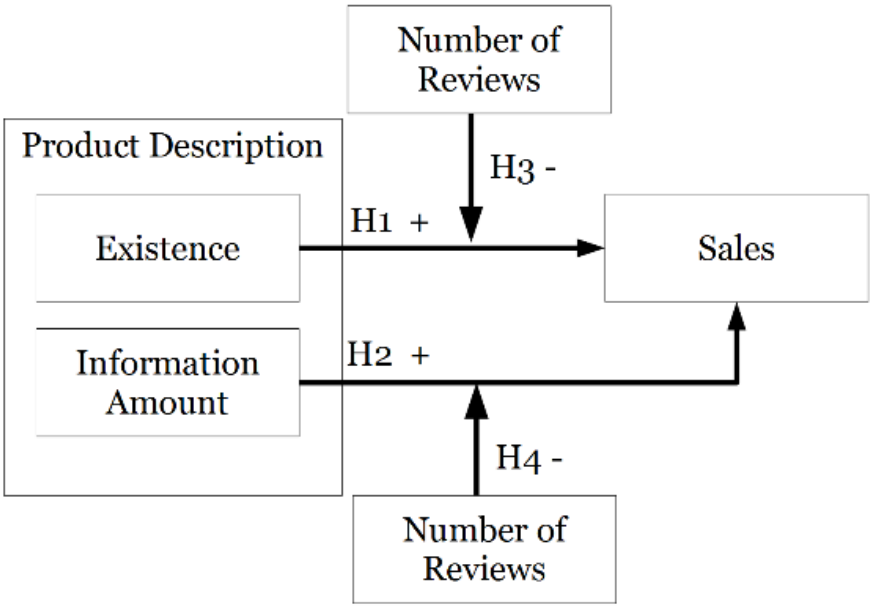

Figure 5.1: Research Model

5.4 Empirical Evaluation

In order to prove our hypotheses, we collected data for three product categories from Amazon.com. We collected data about 84 backpacks, 131 pencils and 136 electric toothbrushes for a period of 39 days. For each product of each category and day, we gathered the price, the average rating and the number of reviews. Amazon provides typically two kinds of product descriptions: descriptions generated by the manufacturer and descriptions generated by the retailer (Amazon.com). Thus, we collected the product description provided by the manufacturer and the product description provided by Amazon.com for each product. Figure 5.2 represents an exemplary product description of an electric toothbrush.

The price range for each product category is rather large as shown in Table 5.1. On the one hand, each category has at least one product which doesn't have any customer reviews (ratings) and on the other hand, each category includes products with a lot of customer reviews (e.g., there are more than 1,000 reviews available for 10 of the electric toothbrushes in our dataset). An Amazon product description is available for 90% of the backpacks, 96% of the pencils and 96% of the electronic toothbrushes in our dataset. Manufacturer product descriptions are significantly less often available than Amazon product descriptions ($p < 0.001$). Only 23% of the collected backpacks, 3% of the pencils and 16% of the electronic toothbrushes provide

Figure 5.2: Exemplary Manufacturer and Amazon Product Description of an Electronic Toothbrush

a manufacturer product description. The mean lengths of Amazon and manufacturer product description differ. Amazon product descriptions furthermore consists of significantly more words than manufacturer descriptions ($p < 0.001$). Amazon descriptions on average are 91.76 words long whereas manufacturer descriptions only consist of 48.13 words. We will use product sales ranks as proxy for sales because sales figures are not available on Amazon.com. The lower the sales rank of a product, the more instances of this product have been sold compared to other products in the same category. The mean sales rank for the backpacks in our dataset is 3911.85, that for pencils is 244.24 and the toothbrushes in our dataset or on average on rank 307.18. For each product category, we collected top sellers (i.e., products with a sales rank of 1) as well as well as niche products (i.e., products with a rather high category-specific sales rank). A summary of the collected data variables is provided in Table 5.1 and Table 5.2.

Table 5.1: Variables Overview

Variable	Symbol	Description
Price	p	Price of the product
Average Rating	avr	Average rating based on customer reviews
Number of Reviews	rev	Number of customer reviews
Amazon Product Description	apd	Availability of an Amazon product description
Manufacturer Product Description	mpd	Availability of a manufacturer product description
Sales Rank	rank	Sales rank of the product
Day	d	Day at which the data have been collected
Length Amazon Product Description	la	Number of words in the Amazon product description
Length manufacturer Product Description	lm	Number of words in the manufacturer product description

Table 5.2: Descriptive Statistics of the Variables in the Dataset

Variable	p	avr	rev	adp	mpd	rank	la	lm
Backpacks (min)	7.99	–	0	0	0	10	0	0
Backpacks (max)	447.5	5	2822	1	1	72690	307	377
Backpacks (mean)	65.82	4.28	388.97	0.91	0.23	3911.85	91.76	48.13
Pencils(min)	1.99	–	0	0	0	1	0	0
Pencils(max)	56.39	5	2588	1	1	1935	322	404
Pencils(mean)	11.15	3.9	91.22	0.96	0.03	244.24	64.43	7.29
Toothbrushes (min)	3.8	–	0	0	0	1	0	0
Toothbrushes (max) 199.99	5	6233	1	1	1466	510	1018	
Toothbrushes (mean)	44.55	3.95	316.47	0.95	0.16	307.18	145	93

5.5 Analysis and Results

5.5.1 Effect of Product Descriptions

In H1, we hypothesized that products with an available product description generate on average more sales than products without a product description. According to H3, the number of customer reviews available for a product moderates this effect. In order to test H1 and H3, we estimate the effect of the presence of a product description binary-coded by the variables apd (Amazon product description) and mpd (manufacturer product description) on product i's sales rank rank at day d. We use price p, average rating avr and number of reviews rev as control variables, because they have been found to significantly affect sales ranks (Zhu and Zhang, 2010). More specifically, we estimate the following model with fixed product and time effects for proofing H1 and H3:

$$log(rank_{i,d}) = \beta_1 log(p_{i,d}) + \beta_2 avr_{i,d} + \beta_3 log(rev_{i,d}+1) + \beta_4 apd_{i,d} + \beta_5 mpd_{i,d} + \beta_6 log(rev_{i,d}+1) apd_{i,d} + \beta_7 log(rev_{i,d}+1) mpd_{i,d} + \beta_8 category + \gamma_1 d + \gamma_2 i + \epsilon_{i,d}$$

We use the logarithm of product i's price to model a diminishing effect of price on sales rank. A price difference between 200 and 210 Euros might be less relevant for a consumer than a price difference between 20 and 30 Euros. We also use the logarithm of each product's number of reviews to model a diminishing effect of this

variable on the sales rank. Furthermore, we use the logarithm of each product's sales rank as dependent variable in order to model diminishing perceived differences between sales ranks (Ho-Dac et al., 2013). Variance inflation factors of less than 2 indicate the absence of multicollinearity in our model. Based on a Durbin-Watson test, we did not find an indication for autocorrelated residuals (D = 1.9998, p = 0.497). The results of our regression model with robust standard errors are depicted in Table 5.3.

Table 5.3: Effect of Product Descriptions on log(Sales Rank)

Variable	Estimate	Std.Error	t-Value	p-Value
log(Price)	0.361	0.030	11.950	< 0.001
Average Rating	0.062	0.016	3.922	< 0.001
log(Number of Reviews+1)	-0.634	0.055	-11.506	< 0.001
Amazon Product Description	-0.778	0.159	-4.898	< 0.001
Manufacturer Product Description	-0.606	0.221	-2.742	0.006
log(Number of Reviews + 1) x Amazon Product Description	0.178	0.051	3.484	< 0.001
log(Number of Reviews + 1) x Manufacturer Product Description	0.083	0.028	2.926	0.003
Pencils	-3.163	0.106	-29.947	< 0.001
Toothbrushes	-2.496	0.134	-18.658	< 0.001
Adj. R^2 (full model)		0.960		
Adj. R^2 (projected model)		0.341		

As expected, Table 5.3 also shows that a lower price and a higher number of customer reviews result in a better sales rank. A somewhat counterintuitive result that emerges from Table 5.3 is that a better (higher) average rating increases the sales rank. One would expect that a better rating would decrease the sales rank. However, previous research has shown that there exists a negative relation between sales and average rating for some product categories (Ghose and Ipeirotis, 2011). Customer ratings typically follow a J-shape distribution with most ratings being very positive (i.e., they are 5-star ratings) (Hu et al., 2017). Consumers might fear manipulated 5-star ratings and rather trust products being characterized not only by 5-star ratings, which ultimately might lead to a positive observed relation between sales rank and a product's average rating. The results in Table 5.3 indicate that both types of product descriptions – Amazon and manufacturer descriptions – significantly influence a product's sales rank. The existence of an Amazon product description on average decreases the sales rank by 2.18 ranks whereas the existence of a manufacturer product description decreases 1.82 ranks. Amazon descriptions hence have a higher impact on a product's sales rank. Manufacturers should consequently create descriptions for their products in order to improve their sales. Our hypothesis H1,

that the presence of a product description has a significantly positive influence on sales is therefore supported by our data.

In H3, we proposed that the effect of product descriptions on sales will diminish with an increasing number of customer reviews. Positive estimates of the interaction effects between the number of reviews and the availability of both, Amazon product descriptions and manufacturer product descriptions, show support for H3.

Our results show a high importance of product descriptions for improving a product's sales. We will analyze the impact of the information amount of Amazon and manufacturer product descriptions in the next section.

5.5.2 Effect of Product Descriptions' Information Amount

In H2, we hypothesized that a higher information amount in a product description will lead to a better sales rank. To proof H2, we use the number of words as a proxy for a description's information amount. H4 hypothesizes an interaction effect between the number of reviews and the information amount of product descriptions.

We use the following model to prove H2 and H4.

$$log(rank_{i,d}) = \beta_1 log(p_{i,d}) + \beta_2 avr_{i,d} + \beta_3 log(rev_{i,d}+1) + \beta_4 log(la_{i,d}+1) + \beta_5 log(lm_{i,d}+1) + \beta_6 log(rev_{i,d}+1) log(la_{i,d}+1) + \beta_7 log(rev_{i,d}+1) log(lm_{i,d}+1) + \beta_8 category + \gamma_1 d + \gamma_2 i + \epsilon_{i,d}$$

With variance inflation factors of less than 2.1, we can assume that our model is not subject to multicollinearity. A Durbin-Watson test also indicated that our model is not subject to an autocorrelation problem (D = 1.9998, p = 0.496). The results in Table 5.4 again show that a lower price and a higher number of customer reviews result in a better sales rank. Table 5.4 also shows that the information amount of Amazon product descriptions has a significant positive influence on a product's sales rank (the higher the information amount, the lower is the sales rank). The effect of manufacturer descriptions is in the same direction but only weakly significant. Hypothesis H2 is hence supported partially.

Table 5.4: Effect of Product Descriptions' Information Amount on log(Sales Rank)

Variable	Estimate	Std.Error	t-Value	p-Value
log(Price)	0.361	0.030	12.097	< 0.001
Average Rating	0.084	0.016	5.252	< 0.001
log(Number of Reviews + 1)	-0.744	0.051	-14.574	< 0.001
log(Length Amazon Description+1)	-0.209	0.030	-7.078	< 0.001
log(Length Manufacturer Description+1)	-0.088	0.048	-1.829	0.068
log(Number of Reviews + 1) x log(Length Amazon Description+1)	0.074	0.011	6.919	< 0.001
log(Number of Reviews + 1) x log(Length Manufacturer Description+1)	0.010	0.006	1.673	0.094
Pencils	-2.746	0.129	-21.218	< 0.001
Toothbrushes	-2.050	0.146	-14.043	< 0.001
Adj. R^2 (full model)		0.960		
Adj. R^2 (full model)		0.344		

H4 proposes a diminishing effect of the information amount of product descriptions on sales when the number of customer reviews increases. As shown in Table 5.4 and Figure 5.3, we found such a diminishing effect for both Amazon and manufacturer product descriptions. The interaction effect of the number of customer reviews and the information amount of manufacturer descriptions is, however, only weakly significant. The more customer reviews are available the higher is the estimate for the information amount indicating that a higher number of available customer reviews leads to a diminishing effect of the information amount on sales. Hypothesis H4 is therefore partially supported by our data.

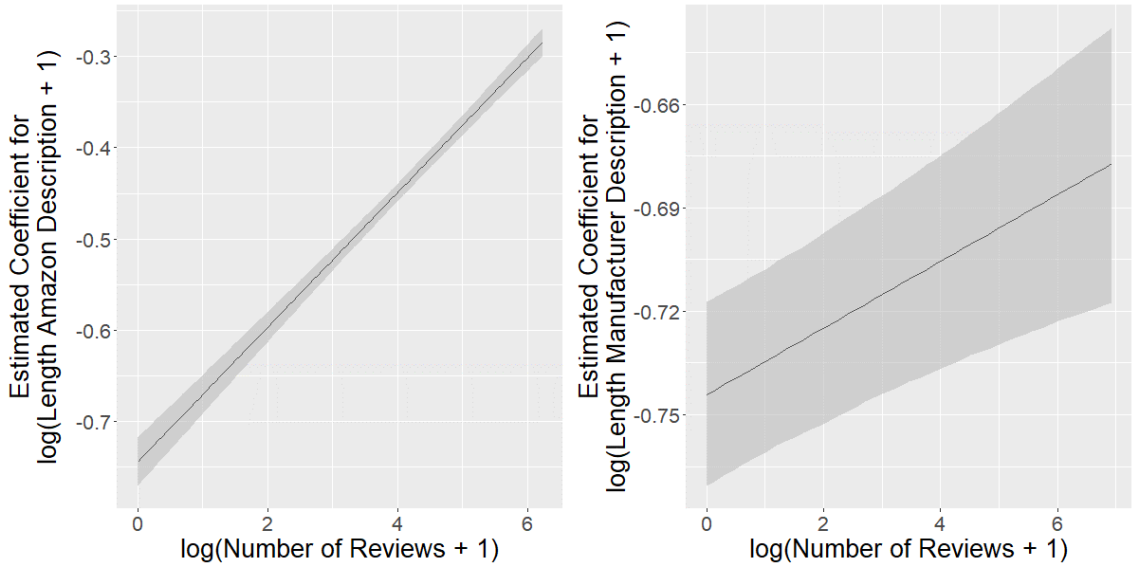

Figure 5.3: Interaction Effect Between the Availability of Product Descriptions and the Number of Reviews

5.6 Robustness Check

We test the robustness of our results by investigating the effects of product descriptions on quantiles rather than average values of sales ranks. We found a very high variance of 13.075.579 for the sales ranks (2.547 for log(sales ranks)) in our data set indicating that our data set encompasses blockbusters as well as niche products. Thus, we analyze the effects of product descriptions on quantiles of sales ranks. More specifically, we run quantile regressions at the 25, 50 and 75% quantile of our dependent variable sales ranks. We estimate the regression model again with fixed product and time effects. Table 5.5 shows the results for the additional quantile regressions and the effect of the availability of product descriptions.

Table 5.5: Results of Quantile Regressions for Availability of Product Descriptions

Variable	Estimate 25% quantile	Estimate 50% quantile	Estimate 75% quantile
log(Price)	0.276***	0.289***	0.284***
Average Rating	-0.002	0.007	0.021
log(Number of Reviews + 1)	-0.530***	-0.553***	-0.562***
Amazon Product Description	-0.552***	-0.511**	-0.487**
Manufacturer Product Description	-0.711***	-0.759***	-0.744***
log(Number of Reviews + 1) x Amazon Product Description	0.104*	0.103*	0.101*
log(Number of Reviews + 1) x Manufacturer Product Description	0.057*	0.062**	0.058**
Pencils	-3.235***	-3.180***	-3.128***
Toothbrushes	-2.576***	-2.601***	-2.627***

Significance codes: *** $p < 0.001$, ** < 0.01, * < 0.05

Table 5.5 indicates that Amazon product descriptions as well as manufacturer descriptions significantly affect sales rank. The influence of manufacturer descriptions is higher than of Amazon descriptions and in opposite to the Amazon descriptions, the influence of manufacturer descriptions slightly increases for niche products. The biggest influence of Amazon product descriptions is for blockbusters. In summary, the effect of the existence of product descriptions is rather robust to products with different sales ranks. Table 5.6 depicts the results for the quantile regressions and the effect of the information amount of product descriptions. All findings from Table 5.4 are supported by the results in Table 5.6. Thus, they are robust for products with different sales ranks.

Table 5.6: Results of Quantile Regressions for Information Amount of Product Descriptions

Variable	Estimate 25% quantile	Estimate 50% quantile	Estimate 75% quantile
log(Price)	0.266***	0.277***	0.273***
Average Rating	0.028	0.037*	0.050**
log(Number of Reviews + 1)	-0.629***	-0.645***	-0.660***
log(Length Amazon Description+1)	-0.168***	-0.150***	-0.156***
log(Length Manufacturer Description+1)	-0.144***	-0.153***	-0.153***
log(Number of Reviews + 1) x log(Length Amazon Description+1)	0.045***	0.043***	0.045***
log(Number of Reviews + 1) x log(Length Manufacturer Description+1)	0.012*	0.013*	0.013*
Pencils	-3.189***	-3.136***	-3.084***
Toothbrushes	-2.550***	-2.582***	-2.607***

Significance codes: *** p < 0.001, ** < 0.01, * < 0.05

5.7 Discussion

This paper examined the influence of product descriptions on sales. Based on empirical data from Amazon.com, we found that the existence of product descriptions positively affects sales. This finding is valid for descriptions generated by the manufacturer as well as descriptions generated by Amazon. Amazon descriptions have been found to have a slightly stronger influence on products' sales than descriptions generated by the manufacturer. Products offered online should hence be described by product descriptions. We furthermore demonstrated that product descriptions that have a higher information amount are more influential on products' sales. The higher the information amount of a description, the better it is prepared to reduce consumers' product uncertainty. We demonstrated that especially Amazon product descriptions positively influence sales rank. Finally, we analyzed the interplay between product descriptions and customer reviews and showed that Amazon-generated product descriptions especially affect sales of products having no or only a few customer reviews. Manufacturer-generated product descriptions, in contrast, have been found to have a higher impact on sales for products having many reviews. This indicates that consumers might have a higher trust in Amazon descriptions. Further, our robustness check showed that the effect of the existence of product descriptions is independent of its sales rank.

Our research hence has three major managerial implications. First, manufacturers and retailers should not only incentivize their consumers to provide reviews but also provide product descriptions on their own. We found that an Amazon-generated product description improves a product's sales rank by more than two positions on average. Second, the longer a product description, the better it helps reducing

product uncertainty and the more consumers finally buy the product. Following Mudambi and Schuff (2010), we assume that longer product descriptions discuss more product characteristics and hence have a higher information amount. And third, it is worthwhile to provide a product description also for products that already got many customer reviews. In the case of Amazon as online store, we found that especially product descriptions generated by manufacturers have a high impact on a product's sales if there are many customer reviews available.

Our research contributes to the ongoing stream of literature on the impact of information sources on consumers' purchase decisions. We demonstrate that the originator of the product description determines the impact of product descriptions on sales. Amazon-generated descriptions have been found to have a higher impact on sales than descriptions generated by the manufacturer.

Our investigation is subject to two major limitations. First, we analyzed the impact of product descriptions and customer reviews on sales only for three product categories – electric toothbrushes, backpacks and pencils. The influence of product descriptions on sales varies across different product categories. Future research should hence investigate further product categories. Second, we found a significant and positive effect of product descriptions on sales, which indicates that consumers use product descriptions to reduce product uncertainty. We, however, did not measure consumers' product uncertainty and instead assumed that product uncertainty has a strong impact on sales. Investigating the effect of product descriptions on consumers' stated product uncertainty and the effect of the stated product uncertainty on sales provides an interesting avenue for future research.

References

Billur Akdeniz, Roger Calantone, and Clay Voorhees. Effectiveness of Marketing Cues on Consumer Perceptions of Quality: The Moderating Roles of Brand Reputation and Third-Party Information. *Psychology and Marketing*, 30:76–89, 2013.

G Akerlof. The Market for " Lemons ": Quality Uncertainty and the Market Mechanism. *The Quarterly Journal of Economics*, 84(3):488–500, 1970.

Nikolay Archak, Anindya Ghose, and Panagiotis G Ipeirotis. Deriving the Pricing Power of Product Features by Mining Consumer Reviews. *Management Science*, 57(8):1485–1509, 2011. ISSN 0025-1909.

Alexander Benlian, Ryad Titah, and Thomas Hess. Differential Effects of Provider Recommendations and Consumer Reviews in E-Commerce Transactions: An Experimental Study. *Journal of Management Information Systems*, 29(1):237–272, 2012. ISSN 0742-1222.

Pei-Yu Chen, Shin-yi Wu, and Jungsun Yoon. The Impact of Online Recommendation and Consumer Feedback on Sales. *Proceeding of the International Conference on Information Systems*, Paper 58:711–724, 2004.

Geng Cui, Hon-Kwong Lui, and Xiaoning Guo. The Effect of Online Consumer Reviews on New Product Sales. *International Journal of Electronic Commerce*, 17(1):39–57, 2012. ISSN 1086-4415.

Brian Detlor, Susan Sproule, and C. Gupta. Pre-purchase online information seeking: Search versus browse. *Journal Electronic Commerce Research*, 4(2):72–84, 2003. URL `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.84.3020{&}rep=rep1{&}type=pdf`.

Angelika Dimoka, Yili Hong, and Paul A Pavlou. On Product Uncertainty in Online Markets: Theory and Evidence. *MIS Quarterly*, 36(2):395–426, 2012.

Verena Dorner, Olga Ivanova, and Michael Scholz. Think Twice Before You Buy! How Recommendations Affect Three-Stage Purchase Decision Processes. *Thirty Fourth International Conference on Information Systems*, 5, 2013.

Wenjing Duan, Bin Gu, and Andrew B. Whinston. Do online reviews matter? - An empirical investigation of panel data. *Decision Support Systems*, 45(4):1007–1016, 2008. ISSN 01679236.

Thomas Ehrmann and Hendrik Schmale. The Hitchhiker's Guide to the Long Tail: The Influence of Online-Reviews and Product Recommendations on Book Sales - Evidence from German Online Retailing. *Icis*, 2008.

Chris Forman, Anindya Ghose, and Batia Wiesenfeld. Examining the relationship between reviews and sales: The role of reviewer identity disclosure in electronic markets. *Information Systems Research*, 19(3):291–313, 2008. ISSN 10477047. doi: 10.1287/isre.1080.0193.

A.a Ghose and S.P.b Han. Estimating demand for mobile applications in the new economy. *Management Science*, 60(6):1470–1488, 2014. ISSN 00251909.

Anindya Ghose. Internet exchanges for used goods: an empirical analysis of trade patterns and adverse selection 1. *MIS Quarterly*, 33(2):263–292, 2009. ISSN 0276-7783.

Anindya Ghose and Panagiotis G Ipeirotis. Estimating the helpfulness and economic impact of product reviews. *International Journal of Innovative Research and Development*, 23(10):1498–1512, 2011. ISSN 1041-4347. doi: 10.1109/TKDE.2010.188. URL http://182.18.175.151/index.php/ijird/article/view/260.

N N Ho-Dac, S J Carson, and W L Moore. The Effects of Positive and Negative Online Customer Reviews: Do Brand Strength and Category Maturity Matter? *Journal of Marketing*, 77(6):37–53, 2013. ISSN 0022-2429.

Yili Kevin Hong and Paul A Pavlou. Product Fit Uncertainty in Online Markets : Nature , Effects , and Antecedents. *Information Systems Research*, 25(2):328–344, 2014. ISSN 1047-7047, 1526-5536.

Nan Hu, Paul A. Pavlou, and Jie Zhang. On Self-Selection Biases in Online Product Reviews. *MIS Quarterly*, 41(2):449–471, 2017. ISSN 02767783. doi: 10.25300/MISQ/2017/41.2.06.

Y Kim and R Krishnan. On product-level uncertainty and online purchase behavior: An empirical analysis. *Management Science*, 61(10):2449–2467, 2015. ISSN 00251909 (ISSN). doi: 10.1287/mnsc.2014.2063.

Ting-Peng Liang and Jin-Shiang Huang. An empirical study on consumer acceptance of products in electronic markets: a transaction cost model. *Decision Support Systems*, 24(1):29–43, 1998.

Yong Liu. Word of Mouth for Movies: Its Dynamics and Impact on Box Office Revenue. *Journal of Marketing*, 70(3):74–89, 2006. ISSN 0022-2429.

Jifeng Luo, Sulin Ba, and Han Zhang. The Effectiveness of online shopping characteristics and well-designed websites on satisfaction. *MIS Quarterly*, 36(4):1131–1144, 2012. ISSN 0276-7783.

Susan Mudambi and David Schuff. What makes a helpful online review? A Study of Customer Reviews on Amazon.com. *MIS Quarterly*, 34(1):185–200, 2010. ISSN 0276-7783.

Jahna Otterbacher. Managing information in online product review communities: A comparison of two approaches. *Ecis*, (2008), 2008. URL http://www.scopus.com/inward/record.url?eid=2-s2.0-84870635276{&}partnerID=tZOtx3y1.

E. Overby and S. Jap. Electronic and Physical Market Channels: A Multiyear Investigation in a Market for Products of Uncertain Quality. *Management Science*, 55(6):940–957, 2009. ISSN 0025-1909.

Paul A. Pavlou, Huigang Liang, and Yajiong Xue. Understanding and mitigating uncertainty in online exchange relationships: a principal- agent perspective. *MIS Quarterly*, 31(1):105–136, 2007.

Nathalia Purnawirawan. Expert Reviewers Beware ! The Effects of Review Set Balance, Review Source and Review Content On Consumer Responses to Online Reviews. *Journal of electronic commerce research*, 15(3):162–178, 2014. ISSN 15266133. URL http://www.jecr.org/sites/default/files/Paper2{_}0.pdf.

Michael Scholz and Verena Dorner. The recipe for the perfect review?: An investigation into the determinants of review helpfulness. *Business and Information Systems Engineering*, 5(3):141–151, 2013. ISSN 18670202.

Feng Zhu and Xiaoquan (Michael) Zhang. Impact of Online Consumer Reviews on Sales : The Moderating Role of Product and Consumer. *Journal of Marketing*, 74 (2):133–148, 2010. ISSN 0022-2429. doi: 10.1509/jmkg.74.2.133.

6 Appendix

A Average classifier ranks for different performance measures for linear relation

Table 1: Average performance measure values for LDLSS

Low Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	9.21	0.000	9.89	0.000	9.89	0.000	9.86	0.000	9.48	0.000	9.49
RDA	10.88	0.000	11.51	0.000	11.51	0.000	11.48	0.000	11.21	0.000	11.24
LR	8.94	0.000	10.02	0.000	10.01	0.000	9.98	0.000	9.07	0.000	9.39
RLR	9.24	0.000	10.10	0.000	10.10	0.000	10.07	0.000	9.55	0.000	9.54
bayLR	8.80	0.000	9.78	0.000	9.78	0.000	9.74	0.000	8.91	0.000	9.17
kNN	16.94	0.000	15.85	0.000	15.85	0.000	15.82	0.000	16.73	0.000	16.36
NSN	11.71	0.000	12.19	0.000	12.19	0.000	12.16	0.000	12.03	0.000	12.00
SNN	14.06	0.000	11.53	0.000	11.53	0.000	11.52	0.000	14.12	0.000	12.43
NB	15.37	0.000	15.02	0.000	15.02	0.000	14.99	0.000	15.67	0.000	15.32
CART	23.30	0.000	18.04	0.000	18.08	0.000	18.36	0.000	22.60	0.000	20.25
C5.0	21.07	0.000	18.93	0.000	18.95	0.000	19.23	0.000	20.54	0.000	20.21
SVM_L	10.48	0.000	11.34	0.000	11.34	0.000	11.31	0.000	10.66	0.000	10.86
SVM_P	11.59	0.000	11.59	0.000	11.59	0.000	11.56	0.000	11.64	0.000	11.56
SVM_R	21.20	0.000	17.94	0.000	17.94	0.000	17.95	0.000	20.67	0.000	19.50
DWD_L	9.78	0.000	10.54	0.000	10.54	0.000	10.53	0.000	9.84	0.000	10.14
DWD_P	11.39	0.000	11.86	0.000	11.86	0.000	11.84	0.000	11.49	0.000	11.80
DWD_R	14.99	0.000	15.88	0.000	15.88	0.000	15.85	0.000	14.91	0.000	15.86
bCART	19.43	0.000	18.00	0.000	18.00	0.000	17.96	0.000	19.59	0.000	18.80
RF	17.38	0.000	16.88	0.000	16.88	0.000	16.82	0.000	17.43	0.000	17.43
booLR	17.88	0.000	18.65	0.000	18.65	0.000	18.63	0.000	17.53	0.000	18.66
GBT	16.52	0.000	16.46	0.000	16.46	0.000	16.43	0.000	16.93	0.000	16.70
SA_A	10.53	0.000	11.16	0.000	11.16	0.000	11.13	0.000	10.59	0.000	10.81
SA_3	**3.75**	–	**6.51**	–	**6.50**	–	**6.50**	–	**3.52**	–	**4.89**
SA_5	4.88	0.000	7.26	0.006	7.25	0.006	7.23	0.006	4.68	0.000	5.89
SA_7	5.70	0.000	8.06	0.000	8.05	0.000	8.03	0.000	5.59	0.000	6.69

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 2: Average performance measure values for LDHSS

Low Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	8.05	0.000	7.11	0.000	7.11	0.000	7.11	0.000	7.93	0.000	7.64
LR	7.67	0.000	7.46	0.000	7.46	0.000	7.46	0.000	7.65	0.000	7.71
RLR	7.25	0.000	7.11	0.000	7.11	0.000	7.11	0.000	7.38	0.000	7.17
bayLR	7.55	0.000	7.29	0.000	7.29	0.000	7.29	0.000	7.54	0.000	7.43
kNN	13.00	0.000	13.04	0.000	13.04	0.000	13.04	0.000	13.00	0.000	13.04
NSN	7.50	0.000	7.09	0.000	7.09	0.000	7.09	0.000	7.65	0.000	7.34
NB	7.62	0.000	7.11	0.000	7.11	0.000	7.11	0.000	7.44	0.000	7.17
CART	14.99	0.000	14.94	0.000	14.94	0.000	14.94	0.000	15.00	0.000	14.94
bCART	**1.00**	–	**1.00**	–	**1.00**	–	**1.00**	–	**1.00**	–	**1.00**
booLR	14.01	0.000	13.80	0.000	13.80	0.000	13.80	0.000	14.00	0.000	13.87
GBT	11.70	0.000	10.53	0.000	10.53	0.000	10.53	0.000	11.74	0.000	11.01
SA_A	5.40	0.000	5.72	0.000	5.72	0.000	5.72	0.000	5.66	0.000	5.51
SA_3	2.93	0.000	4.76	0.000	4.76	0.000	4.76	0.000	2.65	0.000	3.95
SA_5	4.87	0.000	6.12	0.000	6.12	0.000	6.12	0.000	4.86	0.000	5.54
SA_7	6.46	0.000	6.92	0.000	6.92	0.000	6.92	0.000	6.50	0.000	6.67

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 3: Average performance measure values for HDLSS

High Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	14.12	0.000	14.32	0.001	14.34	0.001	14.16	0.007	13.94	0.000	14.44
RDA	14.32	0.000	13.87	0.007	13.89	0.007	14.09	0.008	13.80	0.000	14.44
LR	12.75	0.000	13.88	0.004	13.78	0.004	13.83	0.012	13.03	0.000	13.60
RLR	12.41	0.000	14.65	0.003	14.62	0.003	14.48	0.008	12.99	0.000	13.96
bayLR	12.99	0.000	14.08	0.008	14.08	0.008	14.12	0.034	13.72	0.000	14.24
kNN	15.32	0.000	11.38	0.163	11.41	0.144	11.30	0.431	15.85	0.000	12.46
NSN	13.68	0.000	12.35	0.403	12.36	0.502	12.40	0.414	14.32	0.000	12.94
SNN	13.57	0.000	11.61	0.163	11.57	0.144	11.41	0.339	15.04	0.000	12.51
NB	13.38	0.000	13.69	0.026	13.66	0.026	13.70	0.050	13.27	0.000	13.85
CART	13.80	0.000	11.55	0.171	11.49	0.276	11.58	0.385	14.14	0.000	12.31
C5.0	15.19	0.000	10.49	0.945	10.49	1.000	10.67	1.000	15.69	0.000	11.77
SVM_L	13.43	0.000	13.88	0.001	13.88	0.001	14.02	0.001	13.79	0.000	14.24
SVM_P	14.55	0.000	**9.52**	–	**9.49**	–	**9.73**	–	15.20	0.000	11.16
SVM_R	13.02	0.000	13.05	0.144	13.04	0.144	12.93	0.385	14.43	0.000	13.62
DWD_L	13.79	0.000	14.36	0.001	14.37	0.001	14.38	0.001	14.38	0.000	14.87
DWD_P	14.85	0.000	9.81	0.945	9.80	1.000	9.87	1.000	15.48	0.000	11.35
DWD_R	13.82	0.000	14.59	0.001	14.59	0.001	14.62	0.001	14.53	0.000	15.51
bCART	12.78	0.000	14.62	0.000	14.60	0.000	14.29	0.001	12.77	0.000	14.08
RF	13.82	0.000	13.85	0.002	13.88	0.001	13.99	0.001	13.29	0.000	13.76
booLR	14.83	0.000	13.66	0.003	13.66	0.003	13.62	0.007	14.93	0.000	14.90
GBT	14.43	0.000	13.25	0.163	13.28	0.144	13.46	0.172	13.98	0.000	13.77
SA_A	14.29	0.000	13.27	0.003	13.29	0.002	13.42	0.006	14.41	0.000	14.04
SA_3	**5.10**	–	11.81	0.441	11.82	0.502	11.59	1.000	**2.44**	–	**6.88**
SA_5	6.44	0.002	13.43	0.101	13.46	0.060	13.30	0.140	3.87	0.000	9.17
SA_7	8.34	0.000	14.04	0.003	14.13	0.002	14.06	0.003	5.71	0.000	11.12

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 4: Average performance measure values for HDHSS

High Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	3.63	1.000	4.79	0.000	4.79	0.000	4.74	0.000	3.61	0.016	4.11
LR	3.77	0.466	4.86	0.000	4.86	0.000	4.81	0.000	3.79	0.000	4.57
bayLR	3.85	0.162	4.93	0.000	4.93	0.000	4.88	0.000	3.86	0.000	4.70
NSN	5.80	0.000	6.25	0.000	6.25	0.000	6.21	0.000	5.26	0.000	6.17
NB	7.58	0.000	8.00	0.000	8.00	0.000	7.99	0.000	7.52	0.000	8.07
CART	9.00	0.000	4.13	0.046	4.14	0.072	4.49	0.424	7.09	0.000	5.50
SA_A	4.17	0.162	**1.63**	–	**1.62**	–	**1.57**	–	7.39	0.000	**2.50**
SA_3	**3.48**	–	5.21	0.000	5.21	0.000	5.16	0.000	**2.92**	–	4.68
SA_5	3.71	0.466	5.20	0.000	5.20	0.000	5.15	0.000	3.56	0.001	4.70

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

B Average classifier ranks for different performance measures for quadratic relation

Table 5: Average performance measure values for LDLSS

Low Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	18.00	0.000	18.82	0.000	18.82	0.000	18.82	0.000	17.35	0.000	19.08
RDA	6.25	0.000	9.94	0.000	9.94	0.000	9.94	0.000	5.78	0.000	7.96
LR	17.96	0.000	18.14	0.000	18.14	0.000	18.13	0.000	17.52	0.000	18.47
RLR	17.58	0.000	18.14	0.000	18.14	0.000	18.14	0.000	17.10	0.000	18.29
bayLR	17.77	0.000	17.98	0.000	17.98	0.000	17.98	0.000	17.33	0.000	18.25
kNN	17.00	0.000	**4.77**	–	**4.77**	–	**4.77**	–	20.20	0.000	9.90
NSN	18.26	0.000	19.02	0.000	19.02	0.000	19.00	0.000	17.73	0.000	19.07
SNN	16.84	0.000	8.74	0.000	8.74	0.000	8.74	0.000	18.98	0.000	12.15
NB	8.15	0.000	11.80	0.000	11.80	0.000	11.80	0.000	7.72	0.000	10.11
CART	21.96	0.000	18.97	0.000	18.97	0.000	19.00	0.000	21.36	0.000	20.62
C5.0	13.81	0.000	15.03	0.000	15.03	0.000	15.03	0.000	13.31	0.000	14.64
SVM_L	17.07	0.000	16.27	0.000	16.27	0.000	16.27	0.000	17.21	0.000	16.91
SVM_P	9.70	0.000	7.93	0.000	7.93	0.000	7.93	0.000	10.44	0.000	8.39
SVM_R	11.88	0.000	16.59	0.000	16.59	0.000	16.59	0.000	9.88	0.000	14.56
DWD_L	16.88	0.000	16.10	0.000	16.10	0.000	16.10	0.000	17.15	0.000	16.80
DWD_P	7.00	0.000	6.21	0.012	6.21	0.012	6.21	0.012	7.96	0.000	6.09
DWD_R	4.17	0.000	6.61	0.000	6.61	0.000	6.61	0.000	4.25	0.000	5.04
bCART	13.87	0.000	14.43	0.000	14.43	0.000	14.43	0.000	13.84	0.000	14.50
RF	11.60	0.000	12.96	0.000	12.96	0.000	12.95	0.000	11.42	0.000	12.44
booLR	19.89	0.000	19.61	0.000	19.61	0.000	19.61	0.000	19.36	0.000	20.10
GBT	19.09	0.000	19.00	0.000	19.00	0.000	19.02	0.000	18.77	0.000	19.46
SA_A	10.38	0.000	10.47	0.000	10.47	0.000	10.46	0.000	10.60	0.000	10.26
SA_3	**2.51**	–	5.09	0.249	5.09	0.249	5.09	0.249	**2.38**	–	**3.19**
SA_5	3.24	0.001	6.05	0.001	6.05	0.001	6.05	0.001	3.20	0.000	4.04
SA_7	4.12	0.000	6.33	0.003	6.33	0.003	6.33	0.003	4.17	0.000	4.68

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 6: Average performance measure values for LDHSS

Low Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	14.35	0.000	14.03	0.000	14.03	0.000	14.03	0.000	12.57	0.000	14.34
LR	12.10	0.000	11.52	0.000	11.52	0.000	11.52	0.000	13.02	0.000	11.56
RLR	12.06	0.000	11.47	0.000	11.47	0.000	11.47	0.000	13.03	0.000	11.54
bayLR	12.18	0.000	11.60	0.000	11.60	0.000	11.60	0.000	13.08	0.000	11.68
kNN	6.75	0.000	**1.00**	–	**1.00**	–	**1.00**	–	6.94	0.000	3.46
NSN	14.21	0.000	14.09	0.000	14.09	0.000	14.09	0.000	12.94	0.000	14.38
NB	3.75	0.000	6.14	0.000	6.14	0.000	6.14	0.000	3.32	0.000	5.42
CART	9.31	0.000	12.25	0.000	12.25	0.000	12.25	0.000	9.05	0.000	11.31
bCART	2.31	0.000	2.39	0.000	2.39	0.000	2.39	0.000	2.64	0.000	**1.57**
booLR	9.45	0.000	8.39	0.000	8.39	0.000	8.39	0.000	9.96	0.000	8.67
GBT	6.04	0.000	5.56	0.000	5.56	0.000	5.56	0.000	5.95	0.000	6.24
SA_A	8.34	0.000	8.62	0.000	8.62	0.000	8.62	0.000	8.36	0.000	8.51
SA_3	**1.34**	–	4.09	0.000	4.09	0.000	4.09	0.000	**1.30**	–	2.73
SA_5	2.69	0.000	3.54	0.000	3.54	0.000	3.54	0.000	2.81	0.000	2.96
SA_7	5.13	0.000	5.31	0.000	5.31	0.000	5.31	0.000	5.04	0.000	5.62

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 7: Average performance measure values for HDLSS

High Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	16.05	0.000	15.93	0.000	16.00	0.000	16.43	0.000	14.09	0.000	16.40
RDA	11.23	0.000	14.97	0.000	15.03	0.000	15.45	0.000	7.29	0.000	12.90
LR	15.91	0.000	15.96	0.000	16.11	0.000	16.34	0.000	14.09	0.000	16.50
RLR	14.32	0.000	15.46	0.000	15.44	0.000	15.52	0.000	13.43	0.000	15.80
bayLR	13.18	0.000	15.06	0.000	15.10	0.000	15.03	0.000	12.28	0.000	14.90
kNN	15.69	0.000	**3.20**	–	**3.10**	–	**2.84**	–	23.25	0.000	**7.37**
NSN	13.45	0.000	14.48	0.000	14.52	0.000	14.51	0.000	12.46	0.000	14.56
SNN	16.12	0.000	3.67	0.518	3.60	0.518	3.36	0.518	22.71	0.000	8.12
NB	12.23	0.000	16.55	0.000	16.59	0.000	16.71	0.000	9.80	0.000	14.99
CART	15.57	0.000	14.27	0.000	14.26	0.000	14.45	0.000	13.78	0.000	14.70
C5.0	16.20	0.000	11.86	0.000	11.94	0.000	12.62	0.000	14.65	0.000	13.41
SVM_L	13.09	0.000	15.20	0.000	15.19	0.000	15.12	0.000	12.44	0.000	15.28
SVM_P	13.86	0.000	8.94	0.000	8.80	0.000	8.49	0.000	17.62	0.000	10.66
SVM_R	15.37	0.000	7.26	0.000	7.15	0.000	6.76	0.000	20.55	0.000	10.55
DWD_L	13.30	0.000	15.20	0.000	15.19	0.000	15.26	0.000	12.40	0.000	15.38
DWD_P	14.29	0.000	8.65	0.000	8.58	0.000	8.36	0.000	18.27	0.000	10.60
DWD_R	10.13	0.000	14.59	0.000	14.57	0.000	14.49	0.000	9.45	0.000	13.18
bCART	15.09	0.000	15.16	0.000	15.25	0.000	15.51	0.000	13.00	0.000	15.21
RF	14.18	0.000	13.43	0.000	13.47	0.000	13.69	0.000	13.15	0.000	13.70
booLR	15.01	0.000	15.75	0.000	15.79	0.000	15.86	0.000	12.84	0.000	15.91
GBT	13.95	0.000	15.56	0.000	15.55	0.000	15.44	0.000	12.22	0.000	15.16
SA_A	11.75	0.000	11.88	0.000	11.80	0.000	11.49	0.000	13.95	0.000	11.93
SA_3	4.46	0.821	13.95	0.000	13.98	0.000	13.78	0.000	**2.74**	–	8.80
SA_5	**4.34**	–	13.29	0.000	13.27	0.000	12.93	0.000	3.42	0.011	8.24
SA_7	6.25	0.000	14.71	0.000	14.72	0.000	14.61	0.000	5.12	0.000	10.72

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 8: Average performance measure values for HDHSS

High Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	6.26	0.000	6.11	0.000	6.11	0.000	6.11	0.000	6.12	0.000	6.13
LR	5.76	0.000	5.92	0.000	5.92	0.000	5.92	0.000	5.70	0.000	5.92
bayLR	5.79	0.000	5.96	0.000	5.96	0.000	5.96	0.000	5.72	0.000	5.96
NSN	7.49	0.000	7.31	0.000	7.31	0.000	7.31	0.000	7.40	0.000	7.37
NB	**1.00**	–	**1.00**	–	**1.00**	–	**1.00**	–	**1.00**	–	**1.00**
CART	9.00	0.000	8.45	0.000	8.45	0.000	8.45	0.000	9.00	0.000	8.62
SA_A	2.15	0.000	2.20	0.000	2.20	0.000	2.20	0.000	3.47	0.000	2.21
SA_3	3.92	0.000	4.24	0.000	4.24	0.000	4.24	0.000	3.46	0.000	4.16
SA_5	3.62	0.000	3.81	0.000	3.81	0.000	3.81	0.000	3.13	0.000	3.63

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

C Average classifier ranks for different performance measures for non-normal

Table 9: Average performance measure values for LDLSS

Low Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	10.61	0.000	10.26	0.000	10.24	0.000	10.20	0.000	11.21	0.000	10.27
RDA	10.38	0.000	9.29	0.000	9.28	0.000	9.26	0.000	12.05	0.000	9.55
LR	10.73	0.000	13.61	0.000	13.60	0.000	13.54	0.000	10.29	0.000	12.56
RLR	10.44	0.000	11.66	0.000	11.63	0.000	11.59	0.000	10.64	0.000	11.28
bayLR	10.54	0.000	12.79	0.000	12.77	0.000	12.72	0.000	10.21	0.000	11.86
kNN	16.95	0.000	9.03	0.001	9.03	0.001	9.05	0.001	19.71	0.000	12.65
NSN	11.43	0.000	10.62	0.000	10.61	0.000	10.54	0.000	12.31	0.000	11.07
SNN	18.75	0.000	10.32	0.157	10.31	0.157	10.30	0.157	21.00	0.000	13.88
NB	14.02	0.000	13.57	0.000	13.56	0.000	13.54	0.000	14.85	0.000	13.80
CART	20.79	0.000	17.29	0.000	17.34	0.000	17.59	0.000	19.78	0.000	19.09
C5.0	19.77	0.000	17.96	0.000	18.10	0.000	18.19	0.000	18.18	0.000	18.85
SVM_L	10.90	0.000	11.19	0.000	11.18	0.000	11.13	0.000	11.63	0.000	11.32
SVM_P	11.89	0.000	**5.78**	–	**5.77**	–	**5.74**	–	15.93	0.000	8.17
SVM_R	14.93	0.000	20.18	0.000	20.20	0.000	20.48	0.000	10.48	0.000	17.95
DWD_L	10.57	0.000	12.15	0.000	12.14	0.000	12.07	0.000	10.70	0.000	11.68
DWD_P	12.09	0.000	10.32	0.001	10.31	0.001	10.27	0.001	13.23	0.000	11.06
DWD_R	12.70	0.000	14.57	0.000	14.56	0.000	14.65	0.000	11.60	0.000	13.65
bCART	17.50	0.000	18.54	0.000	18.54	0.000	18.54	0.000	16.38	0.000	18.41
RF	16.91	0.000	17.65	0.000	17.66	0.000	17.61	0.000	15.94	0.000	17.68
booLR	19.61	0.000	20.52	0.000	20.52	0.000	20.43	0.000	17.93	0.000	20.38
GBT	17.90	0.000	18.70	0.000	18.70	0.000	18.67	0.000	16.62	0.000	18.75
SA_A	10.45	0.000	10.71	0.000	10.69	0.000	10.65	0.000	10.96	0.000	10.33
SA_3	**3.88**	–	9.35	0.000	9.34	0.000	9.38	0.000	**3.19**	–	**6.14**
SA_5	5.24	0.000	9.52	0.000	9.51	0.000	9.48	0.000	4.64	0.000	7.15
SA_7	6.04	0.000	9.40	0.000	9.39	0.000	9.37	0.000	5.53	0.000	7.47

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 10: Average performance measure values for LDHSS

Low Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	3.53	0.000	3.14	0.000	3.14	0.000	3.14	0.000	10.68	0.000	3.19
LR	10.21	0.000	11.32	0.000	11.32	0.000	11.32	0.000	4.46	0.000	11.23
RLR	7.84	0.000	8.10	0.000	8.10	0.000	8.10	0.000	7.25	0.000	7.97
bayLR	10.14	0.000	11.20	0.000	11.20	0.000	11.20	0.000	4.43	0.000	11.10
kNN	13.00	0.000	7.39	0.000	7.39	0.000	7.39	0.000	13.42	0.000	10.23
NSN	3.40	0.000	3.17	0.000	3.17	0.000	3.17	0.000	10.58	0.000	3.27
NB	4.28	0.000	2.68	0.000	2.68	0.000	2.68	0.000	12.14	0.000	3.31
CART	14.04	0.000	14.95	0.000	14.95	0.000	14.95	0.000	12.55	0.000	14.61
bCART	**1.00**	–	**1.12**	–	**1.12**	–	**1.12**	–	**1.00**	–	**1.00**
booLR	14.96	0.000	13.74	0.000	13.74	0.000	13.74	0.000	15.00	0.000	14.14
GBT	11.21	0.000	10.31	0.000	10.31	0.000	10.31	0.000	8.54	0.000	10.96
SA_A	4.29	0.000	5.64	0.000	5.64	0.000	5.64	0.000	7.22	0.000	4.76
SA_3	8.82	0.000	10.99	0.000	10.99	0.000	10.99	0.000	3.63	0.000	10.28
SA_5	8.09	0.000	9.34	0.000	9.34	0.000	9.34	0.000	4.00	0.000	8.26
SA_7	5.20	0.000	6.91	0.000	6.91	0.000	6.91	0.000	5.10	0.000	5.71

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 11: Average performance measure values for HDLSS

High Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	13.05	0.000	14.10	0.001	14.12	0.001	13.88	0.008	12.49	0.000	13.71
RDA	14.21	0.000	12.90	0.037	12.93	0.040	13.40	0.011	12.10	0.000	13.00
LR	15.07	0.000	15.02	0.001	15.03	0.000	15.56	0.000	13.22	0.000	15.11
RLR	14.14	0.000	13.20	0.002	13.21	0.002	13.15	0.001	13.73	0.000	13.61
bayLR	14.51	0.000	14.88	0.000	14.88	0.000	14.81	0.000	14.41	0.000	15.14
kNN	12.19	0.000	10.42	0.162	10.35	0.162	9.91	0.240	16.47	0.000	11.12
NSN	13.80	0.000	12.28	0.070	12.24	0.070	12.38	0.033	14.01	0.000	12.99
SNN	14.10	0.000	**8.84**	–	**8.79**	–	**8.71**	–	17.47	0.000	10.66
NB	13.92	0.000	13.36	0.005	13.38	0.002	13.46	0.007	12.71	0.000	13.77
CART	13.44	0.000	11.66	0.162	11.62	0.166	11.92	0.021	12.83	0.000	12.09
C5.0	15.10	0.000	11.10	0.175	11.07	0.256	11.10	0.240	15.48	0.000	12.79
SVM_L	14.22	0.000	14.72	0.000	14.70	0.000	14.71	0.000	13.99	0.000	15.18
SVM_P	13.94	0.000	11.98	0.162	11.97	0.129	11.96	0.060	16.29	0.000	13.04
SVM_R	14.03	0.000	12.29	0.000	12.29	0.000	12.16	0.001	15.77	0.000	13.16
DWD_L	14.39	0.000	14.06	0.000	14.11	0.000	14.18	0.001	14.13	0.000	14.93
DWD_P	14.04	0.000	10.72	0.175	10.69	0.256	10.58	0.240	15.85	0.000	11.73
DWD_R	14.13	0.000	13.76	0.009	13.79	0.003	13.93	0.002	12.61	0.000	13.74
bCART	14.23	0.000	13.88	0.011	13.93	0.008	14.04	0.005	13.62	0.000	14.27
RF	13.03	0.000	13.94	0.000	13.93	0.000	13.84	0.001	12.98	0.000	13.79
booLR	13.34	0.000	13.96	0.000	14.04	0.000	13.94	0.000	13.88	0.000	14.18
GBT	14.14	0.000	14.84	0.000	14.81	0.000	15.08	0.000	14.04	0.000	15.38
SA_A	13.73	0.000	11.87	0.037	11.82	0.046	11.74	0.013	14.89	0.000	12.65
SA_3	**5.16**	–	13.66	0.000	13.69	0.000	13.38	0.000	**2.89**	–	**8.94**
SA_5	5.77	0.258	13.30	0.002	13.28	0.002	13.07	0.006	3.70	0.078	9.14
SA_7	7.32	0.000	14.28	0.000	14.31	0.000	14.07	0.001	5.42	0.000	10.89

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 12: Average performance measure values for HDHSS

High Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	3.60	1.000	4.24	0.000	4.24	0.000	4.24	0.000	5.14	0.000	3.29
LR	3.67	0.610	7.13	0.000	7.13	0.000	7.13	0.000	2.45	0.412	6.82
bayLR	3.61	1.000	7.09	0.000	7.09	0.000	7.09	0.000	2.42	0.705	6.81
NSN	5.38	0.000	6.32	0.000	6.32	0.000	6.32	0.000	5.28	0.000	6.03
NB	7.86	0.000	**1.76**	–	**1.76**	–	**1.76**	–	8.13	0.000	3.65
CART	9.00	0.000	2.71	0.000	2.71	0.000	2.71	0.000	8.31	0.000	3.73
SA_A	4.40	0.059	2.79	0.000	2.79	0.000	2.79	0.000	7.03	0.000	**2.77**
SA_3	**3.56**	–	7.09	0.000	7.09	0.000	7.09	0.000	**2.37**	–	6.78
SA_5	3.92	0.360	5.88	0.000	5.88	0.000	5.88	0.000	3.88	0.000	5.12

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

D Average classifier ranks for different performance measures for multimodal

Table 13: Average performance measure values for LDLSS

Low Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	16.16	0.000	15.08	0.000	15.07	0.000	15.10	0.000	16.55	0.000	16.52
RDA	9.23	0.000	12.06	0.000	12.04	0.000	11.55	0.000	11.03	0.000	11.10
LR	15.83	0.000	15.05	0.000	15.05	0.000	15.13	0.000	16.04	0.000	16.32
RLR	15.49	0.000	15.04	0.000	14.99	0.000	15.11	0.000	16.30	0.000	15.84
bayLR	15.88	0.000	15.36	0.000	15.43	0.000	15.50	0.000	16.21	0.000	16.71
kNN	11.62	0.000	13.73	0.000	13.71	0.000	13.93	0.000	9.46	0.000	12.05
NSN	15.94	0.000	14.32	0.000	14.36	0.000	14.25	0.000	17.56	0.000	16.18
SNN	13.43	0.000	12.04	0.000	12.05	0.000	12.54	0.000	10.90	0.000	12.13
NB	13.22	0.000	14.26	0.000	14.30	0.000	14.12	0.000	13.62	0.000	14.32
CART	16.11	0.000	9.68	0.000	9.69	0.000	9.91	0.000	15.40	0.000	11.18
C5.0	16.88	0.000	**4.95**	–	**5.00**	–	**5.50**	–	16.30	0.000	7.96
SVM_L	15.47	0.000	14.54	0.000	14.56	0.000	14.63	0.000	14.74	0.000	15.28
SVM_P	12.97	0.000	12.45	0.000	12.47	0.000	12.54	0.000	11.85	0.000	12.04
SVM_R	11.18	0.000	9.39	0.000	9.33	0.000	8.89	0.009	14.48	0.000	9.96
DWD_L	15.91	0.000	14.65	0.000	14.68	0.000	14.74	0.000	15.29	0.000	15.54
DWD_P	10.09	0.000	13.35	0.000	13.30	0.000	13.03	0.000	10.48	0.000	11.84
DWD_R	9.61	0.000	13.56	0.000	13.52	0.000	13.29	0.000	10.69	0.000	12.29
bCART	14.02	0.000	14.22	0.000	14.21	0.000	14.30	0.000	14.69	0.000	14.78
RF	12.93	0.000	13.03	0.000	12.99	0.000	12.91	0.000	14.38	0.000	13.40
booLR	16.94	0.000	13.93	0.000	13.94	0.000	13.96	0.000	16.68	0.000	15.44
GBT	15.94	0.000	13.34	0.000	13.27	0.000	13.28	0.000	17.11	0.000	14.77
SA_A	13.51	0.000	12.95	0.000	13.02	0.000	13.03	0.000	13.41	0.000	13.01
SA_3	**5.12**	–	11.04	0.000	11.03	0.000	10.99	0.000	**2.94**	–	**7.20**
SA_5	5.68	0.564	13.11	0.000	13.11	0.000	12.88	0.000	3.90	0.029	8.93
SA_7	5.83	0.005	13.88	0.000	13.89	0.000	13.90	0.000	5.02	0.000	10.21

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 14: Average performance measure values for LDHSS

Low Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	12.96	0.000	12.86	0.000	12.86	0.000	12.86	0.000	12.96	0.000	12.98
LR	12.88	0.000	12.89	0.000	12.89	0.000	12.89	0.000	12.84	0.000	12.89
RLR	13.06	0.000	12.90	0.000	12.90	0.000	12.90	0.000	13.01	0.000	12.93
bayLR	13.00	0.000	12.89	0.000	12.89	0.000	12.89	0.000	12.91	0.000	12.94
kNN	3.39	0.000	3.61	0.000	3.51	0.000	2.96	0.000	5.24	0.000	3.07
NSN	13.09	0.000	13.00	0.000	13.00	0.000	13.00	0.000	13.28	0.000	13.05
NB	7.49	0.000	8.55	0.000	8.55	0.000	8.55	0.000	8.93	0.000	8.51
CART	8.96	0.000	9.57	0.000	9.57	0.000	9.58	0.000	9.66	0.000	9.61
bCART	**1.00**	–	**1.11**	–	**1.06**	–	**1.00**	–	**1.00**	–	**1.00**
booLR	10.00	0.000	2.78	0.000	3.05	0.000	4.76	0.000	7.02	0.000	5.55
GBT	5.03	0.000	5.58	0.000	5.57	0.000	5.32	0.000	4.60	0.000	5.29
SA_A	7.55	0.000	9.33	0.000	9.33	0.000	9.32	0.000	8.39	0.000	9.10
SA_3	2.00	0.000	2.63	0.000	2.53	0.000	2.04	0.000	2.00	0.000	2.00
SA_5	3.61	0.000	5.31	0.000	5.30	0.000	5.07	0.000	3.00	0.000	4.25
SA_7	5.97	0.000	6.98	0.000	6.98	0.000	6.85	0.000	5.16	0.000	6.84

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 15: Average performance measure values for HDLSS

High Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	15.36	0.000	13.15	0.075	13.11	0.080	13.46	0.007	15.32	0.000	14.80
RDA	12.88	0.000	13.74	0.001	13.69	0.001	13.63	0.003	13.22	0.000	13.45
LR	13.80	0.000	14.12	0.085	14.09	0.097	13.96	0.056	14.31	0.000	14.49
RLR	13.18	0.000	13.31	0.084	13.30	0.111	13.10	0.064	13.17	0.000	13.54
bayLR	13.66	0.000	15.12	0.001	15.09	0.001	14.86	0.002	13.66	0.000	15.05
kNN	13.62	0.000	11.61	1.000	11.64	1.000	11.72	1.000	14.37	0.000	12.05
NSN	14.66	0.000	14.03	0.004	14.12	0.004	14.24	0.009	14.35	0.000	14.49
SNN	14.04	0.000	11.27	1.000	11.29	1.000	11.29	1.000	14.83	0.000	11.99
NB	13.94	0.000	13.27	0.306	13.34	0.306	13.66	0.165	13.72	0.000	13.77
CART	14.39	0.000	11.53	0.950	11.49	0.950	11.56	1.000	14.47	0.000	12.60
C5.0	14.03	0.000	11.95	1.000	11.98	1.000	12.04	1.000	15.47	0.000	12.90
SVM_L	13.52	0.000	14.03	0.028	13.98	0.021	13.95	0.009	13.13	0.000	13.96
SVM_P	13.11	0.000	11.26	1.000	11.23	1.000	11.13	1.000	13.95	0.000	11.44
SVM_R	13.62	0.000	13.64	0.197	13.69	0.131	13.60	0.191	14.73	0.000	14.10
DWD_L	13.07	0.000	14.39	0.028	14.35	0.027	14.39	0.027	12.96	0.000	14.28
DWD_P	13.46	0.000	**10.28**	–	**10.30**	–	**10.36**	–	13.49	0.000	11.04
DWD_R	13.50	0.000	14.08	0.028	14.04	0.046	13.91	0.031	13.23	0.000	14.27
bCART	14.70	0.000	14.59	0.054	14.60	0.097	14.54	0.152	15.02	0.000	15.65
RF	13.88	0.000	13.11	0.660	13.12	0.875	13.04	0.875	14.11	0.000	13.48
booLR	15.45	0.000	12.07	1.000	12.12	1.000	12.49	0.925	16.04	0.000	13.81
GBT	14.80	0.000	13.51	0.056	13.49	0.046	13.36	0.056	15.37	0.000	14.51
SA_A	13.23	0.000	13.38	0.271	13.38	0.271	13.27	0.408	13.73	0.000	13.52
SA_3	**5.14**	–	11.64	0.660	11.62	0.689	11.54	0.689	**2.90**	–	**7.17**
SA_5	6.57	0.002	12.72	0.160	12.76	0.147	12.70	0.160	4.08	0.001	8.94
SA_7	7.39	0.000	13.20	0.021	13.16	0.021	13.18	0.047	5.36	0.000	9.72

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 16: Average performance measure values for HDHSS

High Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	5.37	0.000	6.50	0.000	6.50	0.000	6.50	0.000	4.82	0.000	6.32
LR	5.42	0.000	6.45	0.000	6.45	0.000	6.45	0.000	4.89	0.000	6.38
bayLR	5.42	0.000	6.47	0.000	6.47	0.000	6.47	0.000	4.86	0.000	6.44
NSN	6.61	0.000	7.39	0.000	7.39	0.000	7.39	0.000	5.61	0.000	7.13
NB	**1.55**	–	3.32	0.000	3.31	0.000	3.30	0.000	**1.99**	–	**2.03**
CART	8.45	0.000	**1.27**	–	**1.27**	–	**1.27**	–	7.25	0.000	3.46
SA_A	3.55	0.000	2.55	0.000	2.55	0.000	2.55	0.000	8.17	0.000	3.06
SA_3	3.45	0.000	4.82	0.000	4.83	0.000	4.84	0.000	2.86	0.000	4.07
SA_5	5.17	0.000	6.24	0.000	6.24	0.000	6.24	0.000	4.57	0.000	6.11

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

E Average classifier ranks for different performance measures for unequal weights

Table 17: Average performance measure values for LDLSS

Low Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	9.66	0.000	10.51	0.000	10.51	0.000	10.51	0.000	9.77	0.000	10.01
RDA	12.33	0.000	12.66	0.000	12.66	0.000	12.66	0.000	12.56	0.000	12.39
LR	9.09	0.000	10.05	0.000	10.05	0.000	10.05	0.000	9.21	0.000	9.47
RLR	10.55	0.000	11.09	0.000	11.09	0.000	11.09	0.000	10.61	0.000	10.84
bayLR	9.55	0.000	10.35	0.000	10.35	0.000	10.34	0.000	9.67	0.000	9.90
kNN	18.61	0.000	17.08	0.000	17.08	0.000	17.08	0.000	18.57	0.000	18.16
NSN	13.22	0.000	13.17	0.000	13.17	0.000	13.16	0.000	13.25	0.000	13.25
SNN	15.39	0.000	14.12	0.000	14.12	0.000	14.12	0.000	15.64	0.000	15.04
NB	16.20	0.000	15.77	0.000	15.77	0.000	15.76	0.000	16.16	0.000	16.23
CART	19.69	0.000	16.20	0.000	16.20	0.000	16.20	0.000	19.48	0.000	17.79
C5.0	18.53	0.000	17.66	0.000	17.66	0.000	17.66	0.000	18.05	0.000	18.25
SVM_L	10.39	0.000	11.04	0.000	11.04	0.000	11.04	0.000	10.85	0.000	10.72
SVM_P	12.65	0.000	12.99	0.000	12.98	0.000	12.98	0.000	12.83	0.000	12.95
SVM_R	21.77	0.000	18.82	0.000	18.83	0.000	18.85	0.000	21.25	0.000	20.09
DWD_L	9.93	0.000	10.52	0.000	10.52	0.000	10.52	0.000	9.96	0.000	10.29
DWD_P	11.04	0.000	11.72	0.000	11.72	0.000	11.72	0.000	11.27	0.000	11.50
DWD_R	15.13	0.000	14.79	0.000	14.78	0.000	14.78	0.000	15.33	0.000	15.24
bCART	16.20	0.000	14.91	0.000	14.91	0.000	14.91	0.000	16.62	0.000	15.64
RF	15.80	0.000	15.13	0.000	15.13	0.000	15.13	0.000	16.05	0.000	15.54
booLR	17.18	0.000	17.16	0.000	17.16	0.000	17.16	0.000	16.81	0.000	17.11
GBT	17.29	0.000	16.99	0.000	16.99	0.000	16.99	0.000	16.86	0.000	17.25
SA_A	10.32	0.000	10.84	0.000	10.84	0.000	10.84	0.000	10.52	0.000	10.49
SA_3	**3.69**	–	**6.25**	–	**6.25**	–	**6.25**	–	**3.46**	–	**4.50**
SA_5	5.07	0.000	7.41	0.003	7.41	0.003	7.41	0.003	4.80	0.000	5.93
SA_7	5.71	0.000	7.78	0.000	7.78	0.000	7.78	0.000	5.41	0.000	6.45

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 18: Average performance measure values for LDHSS

Low Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	6.71	0.000	6.12	0.000	6.12	0.000	6.12	0.000	6.81	0.000	6.43
LR	6.55	0.000	6.37	0.000	6.37	0.000	6.37	0.000	6.54	0.000	6.54
RLR	7.01	0.000	6.97	0.000	6.97	0.000	6.97	0.000	6.92	0.000	7.15
bayLR	6.50	0.000	6.37	0.000	6.37	0.000	6.37	0.000	6.50	0.000	6.41
kNN	13.00	0.000	13.38	0.000	13.38	0.000	13.38	0.000	13.00	0.000	13.27
NSN	10.97	0.000	10.91	0.000	10.91	0.000	10.91	0.000	10.97	0.000	10.94
NB	12.00	0.000	12.15	0.000	12.15	0.000	12.15	0.000	11.99	0.000	12.12
CART	14.83	0.000	14.23	0.000	14.23	0.000	14.23	0.000	14.75	0.000	14.46
bCART	**1.00**	–	**1.02**	–	**1.02**	–	**1.02**	–	**1.00**	–	**1.00**
booLR	14.17	0.000	13.03	0.000	13.03	0.000	13.03	0.000	14.25	0.000	13.55
GBT	9.52	0.000	8.82	0.000	8.82	0.000	8.82	0.000	9.39	0.000	9.11
SA_A	2.62	0.000	3.82	0.000	3.82	0.000	3.82	0.000	3.00	0.000	3.26
SA_3	3.98	0.000	5.29	0.000	5.29	0.000	5.29	0.000	3.77	0.000	4.57
SA_5	5.92	0.000	6.12	0.000	6.12	0.000	6.12	0.000	5.78	0.000	6.00
SA_7	5.21	0.000	5.40	0.000	5.40	0.000	5.40	0.000	5.32	0.000	5.20

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 19: Average performance measure values for HDLSS

High Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	13.87	0.000	12.11	0.940	12.16	0.940	12.48	1.000	13.12	0.000	12.29
RDA	14.84	0.000	12.93	0.017	13.00	0.010	13.19	0.007	13.71	0.000	13.67
LR	15.13	0.000	13.65	0.312	13.61	0.339	13.65	0.129	14.68	0.000	14.22
RLR	14.80	0.000	14.45	0.002	14.49	0.001	14.60	0.001	14.66	0.000	15.52
bayLR	14.74	0.000	13.97	0.005	13.99	0.002	14.01	0.002	14.68	0.000	14.57
kNN	13.38	0.000	11.80	0.940	11.74	0.940	11.73	1.000	14.86	0.000	12.57
NSN	14.99	0.000	12.76	0.515	12.76	0.644	12.97	1.000	14.64	0.000	13.79
SNN	13.93	0.000	11.56	0.646	11.56	0.939	11.77	1.000	14.30	0.000	12.34
NB	14.41	0.000	13.27	0.091	13.27	0.091	13.33	0.245	13.97	0.000	13.52
CART	13.90	0.000	**9.73**	–	**9.73**	–	**9.91**	–	15.19	0.000	10.89
C5.0	13.28	0.000	12.49	0.126	12.43	0.306	12.18	1.000	14.12	0.000	12.54
SVM_L	13.71	0.000	13.54	0.005	13.53	0.010	13.49	0.033	13.52	0.000	13.79
SVM_P	13.48	0.000	11.70	0.940	11.70	0.940	11.63	1.000	14.30	0.000	12.25
SVM_R	13.82	0.000	12.41	0.206	12.42	0.339	12.39	0.540	14.21	0.000	12.75
DWD_L	13.45	0.000	13.43	0.085	13.44	0.085	13.45	0.045	13.61	0.000	13.77
DWD_P	13.65	0.000	11.98	0.905	11.88	0.939	11.81	1.000	14.61	0.000	12.38
DWD_R	13.81	0.000	13.94	0.010	13.97	0.010	14.04	0.011	13.05	0.000	14.05
bCART	12.05	0.000	13.66	0.009	13.62	0.012	13.46	0.014	12.93	0.000	13.29
RF	14.48	0.000	12.85	0.085	12.89	0.085	12.88	0.101	15.60	0.000	14.21
booLR	15.61	0.000	14.61	0.005	14.63	0.007	14.90	0.002	15.12	0.000	15.31
GBT	14.54	0.000	13.44	0.001	13.44	0.000	13.52	0.005	14.30	0.000	14.24
SA_A	14.04	0.000	13.35	0.005	13.38	0.006	13.55	0.003	14.15	0.000	14.14
SA_3	**4.24**	–	12.91	0.085	12.91	0.085	12.44	0.203	**2.98**	–	**8.69**
SA_5	4.76	0.009	13.62	0.001	13.61	0.001	13.14	0.020	3.70	0.002	9.06
SA_7	6.08	0.000	14.86	0.000	14.82	0.000	14.47	0.001	4.97	0.000	11.14

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 20: Average performance measure values for HDHSS

High Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	4.25	0.070	4.71	0.000	4.71	0.000	4.70	0.000	3.81	0.009	4.43
LR	4.04	0.028	4.58	0.000	4.58	0.000	4.57	0.000	3.46	0.005	4.43
bayLR	4.04	0.058	4.57	0.000	4.57	0.000	4.55	0.000	3.46	0.008	4.38
NSN	4.80	0.011	5.55	0.000	5.55	0.000	5.53	0.000	4.22	0.008	5.11
NB	7.57	0.000	7.32	0.000	7.32	0.000	7.30	0.000	7.39	0.000	7.53
CART	9.00	0.000	8.27	0.000	8.27	0.000	8.43	0.000	8.44	0.000	8.51
SA_A	3.60	0.841	**1.09**	–	**1.09**	–	**1.07**	–	7.45	0.000	**2.21**
SA_3	**3.53**	–	4.32	0.000	4.32	0.000	4.30	0.000	**2.97**	–	3.98
SA_5	4.17	0.025	4.58	0.000	4.58	0.000	4.56	0.000	3.80	0.005	4.42

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

F Average classifier ranks for different performance measures for unbalanced

Table 21: Average performance measure values for LDLSS

Low Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	9.85	0.000	15.76	0.000	15.76	0.000	15.76	0.000	8.30	0.000	13.82
RDA	11.09	0.000	14.09	0.000	14.09	0.000	14.09	0.000	10.77	0.000	12.91
LR	10.03	0.000	17.23	0.000	17.23	0.000	17.23	0.000	7.42	0.000	14.31
RLR	9.54	0.000	13.20	0.000	13.20	0.000	13.20	0.000	9.46	0.000	11.34
bayLR	9.34	0.000	14.46	0.000	14.46	0.000	14.46	0.000	8.54	0.000	11.92
kNN	14.43	0.000	8.64	0.000	8.64	0.000	8.64	0.000	17.95	0.000	10.89
NSN	17.97	0.000	**4.84**	–	**4.84**	–	**4.84**	–	23.05	0.000	10.61
SNN	18.16	0.000	10.08	0.000	10.08	0.000	10.08	0.000	20.52	0.000	13.89
NB	14.05	0.000	13.54	0.000	13.54	0.000	13.54	0.000	14.39	0.000	14.23
CART	21.19	0.000	18.10	0.000	18.10	0.000	18.10	0.000	17.20	0.000	19.66
C5.0	18.95	0.000	15.46	0.000	15.46	0.000	15.46	0.000	16.98	0.000	17.33
SVM_L	10.21	0.000	14.01	0.000	14.01	0.000	14.01	0.000	10.04	0.000	12.36
SVM_P	13.12	0.000	8.64	0.000	8.64	0.000	8.64	0.000	16.55	0.000	10.26
SVM_R	18.04	0.000	4.96	0.157	4.96	0.157	4.96	0.157	23.02	0.000	10.85
DWD_L	10.63	0.000	13.95	0.000	13.95	0.000	13.95	0.000	9.90	0.000	12.51
DWD_P	12.46	0.000	12.63	0.000	12.63	0.000	12.63	0.000	12.62	0.000	12.31
DWD_R	14.47	0.000	10.52	0.000	10.52	0.000	10.52	0.000	16.85	0.000	12.27
bCART	17.55	0.000	18.05	0.000	18.05	0.000	18.05	0.000	14.81	0.000	19.03
RF	14.95	0.000	13.07	0.000	13.07	0.000	13.07	0.000	15.51	0.000	14.26
booLR	16.66	0.000	20.59	0.000	20.59	0.000	20.59	0.000	10.96	0.000	19.56
GBT	14.27	0.000	15.69	0.000	15.69	0.000	15.69	0.000	13.30	0.000	15.94
SA_A	10.70	0.000	9.62	0.000	9.62	0.000	9.62	0.000	12.90	0.000	9.41
SA_3	**4.80**	–	12.37	0.000	12.37	0.000	12.37	0.000	**3.40**	–	**7.57**
SA_5	5.84	0.002	12.40	0.000	12.40	0.000	12.40	0.000	4.80	0.000	8.26
SA_7	6.70	0.000	13.11	0.000	13.11	0.000	13.11	0.000	5.73	0.000	9.52

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 22: Average performance measure values for LDHSS

Low Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	5.92	0.000	6.04	0.000	6.04	0.000	6.04	0.000	8.02	0.000	4.75
LR	6.79	0.000	12.05	0.000	12.05	0.000	12.05	0.000	3.03	0.000	11.65
RLR	6.77	0.000	9.75	0.000	9.75	0.000	9.75	0.000	5.79	0.000	10.24
bayLR	7.06	0.000	11.83	0.000	11.83	0.000	11.83	0.000	3.81	0.000	11.76
kNN	11.00	0.000	13.64	0.000	13.64	0.000	13.64	0.000	11.15	0.000	13.95
NSN	14.37	0.000	**1.49**	–	**1.49**	–	**1.49**	–	14.37	0.000	4.31
NB	12.00	0.000	2.98	0.000	2.98	0.000	2.98	0.000	13.00	0.000	6.01
CART	14.63	0.000	1.56	0.157	1.56	0.157	1.56	0.157	14.63	0.000	5.27
bCART	**1.00**	–	6.96	0.000	6.96	0.000	6.96	0.000	**1.00**	–	**1.30**
booLR	13.00	0.000	15.00	0.000	15.00	0.000	15.00	0.000	11.67	0.000	14.99
GBT	9.68	0.000	4.97	0.000	4.97	0.000	4.97	0.000	9.97	0.000	7.29
SA_A	2.85	0.000	4.01	0.000	4.01	0.000	4.01	0.000	9.18	0.000	1.73
SA_3	6.42	0.000	11.54	0.000	11.54	0.000	11.54	0.000	3.06	0.000	11.21
SA_5	5.32	0.000	10.12	0.000	10.12	0.000	10.12	0.000	4.63	0.000	9.73
SA_7	3.21	0.000	8.05	0.000	8.05	0.000	8.05	0.000	6.68	0.000	5.80

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 23: Average performance measure values for HDLSS

High Dimensionality - Low Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	10.53	0.000	9.79	0.124	9.78	0.124	9.78	0.124	16.02	0.000	9.68
RDA	15.90	0.000	13.10	0.000	13.06	0.000	13.04	0.000	12.82	0.000	16.23
LR	24.86	0.000	23.12	0.000	23.70	0.000	24.39	0.000	**3.42**	–	23.86
RLR	20.05	0.000	19.39	0.000	19.32	0.000	19.21	0.000	8.20	0.000	21.19
bayLR	14.44	0.000	11.96	0.000	11.94	0.000	11.93	0.000	13.34	0.000	14.38
kNN	11.31	0.000	10.10	0.048	10.10	0.048	10.10	0.048	15.81	0.000	10.56
NSN	8.78	0.073	9.54	0.327	9.54	0.327	9.54	0.327	16.32	0.000	8.07
SNN	8.06	0.635	9.43	0.635	9.43	0.635	9.43	0.635	16.86	0.000	7.22
NB	8.99	0.073	9.44	0.635	9.44	0.635	9.44	0.635	16.17	0.000	8.14
CART	21.02	0.000	20.42	0.000	20.42	0.000	20.29	0.000	7.75	0.000	21.25
C5.0	14.76	0.000	15.29	0.000	15.28	0.000	15.18	0.000	11.41	0.000	15.19
SVM_L	17.77	0.000	15.72	0.000	15.64	0.000	15.59	0.000	10.41	0.000	18.46
SVM_P	9.37	0.004	10.29	0.032	10.28	0.032	10.27	0.032	16.10	0.000	8.67
SVM_R	**8.01**	–	9.35	–	9.35	–	9.35	–	16.98	0.000	**7.12**
DWD_L	11.31	0.000	11.16	0.001	11.14	0.001	11.13	0.001	15.26	0.000	10.77
DWD_P	11.49	0.000	11.24	0.001	11.22	0.001	11.21	0.001	14.92	0.000	11.05
DWD_R	8.06	0.635	**9.35**	–	**9.35**	–	**9.35**	–	16.98	0.000	7.18
bCART	12.24	0.000	10.74	0.003	10.72	0.003	10.72	0.003	14.64	0.000	12.01
RF	8.41	0.327	9.59	0.466	9.59	0.466	9.58	0.466	16.75	0.000	7.57
booLR	18.75	0.000	17.55	0.000	17.48	0.000	17.41	0.000	9.04	0.000	19.50
GBT	13.60	0.000	10.49	0.007	10.48	0.007	10.47	0.007	15.35	0.000	12.70
SA_A	8.17	0.327	9.35	–	9.35	–	9.35	–	16.98	0.000	7.29
SA_3	15.54	0.000	19.50	0.000	19.43	0.000	19.34	0.000	4.70	0.264	19.14
SA_5	12.62	0.000	16.06	0.000	15.98	0.000	15.95	0.000	7.71	0.000	15.60
SA_7	10.97	0.009	13.05	0.000	13.01	0.000	12.98	0.000	11.07	0.000	12.15

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.

Table 24: Average performance measure values for HDHSS

High Dimensionality - High Sample Size											
	PCC	p-value	AUC	p-value	H	p-value	BS	p-value	F1	p-value	AvgR
LDA	4.96	1.000	5.58	0.000	5.58	0.000	5.58	0.000	4.20	0.000	5.56
LR	5.03	1.000	5.53	0.000	5.53	0.000	5.53	0.000	4.26	0.000	5.52
bayLR	4.99	1.000	5.50	0.000	5.50	0.000	5.50	0.000	4.24	0.000	5.49
NSN	4.61	–	2.44	–	2.44	–	2.44	–	7.99	0.000	2.73
NB	6.25	0.003	8.47	0.000	8.47	0.000	8.47	0.000	**2.15**	–	8.21
CART	**4.61**	–	**2.44**	–	**2.44**	–	**2.44**	–	7.99	0.000	**2.73**
SA_A	4.87	1.000	4.08	0.000	4.08	0.000	4.08	0.000	5.76	0.000	4.07
SA_3	4.92	1.000	5.46	0.000	5.46	0.000	5.46	0.000	4.17	0.000	5.42
SA_5	4.77	1.000	5.50	0.000	5.50	0.000	5.50	0.000	4.25	0.000	5.29

Bold face indicates the best classifier (lowest rank) per performance measure. *p-value* are the adjusted p-values of each performance measure to a pairwise comparison of the best classifier (per performance measure) to each other.